

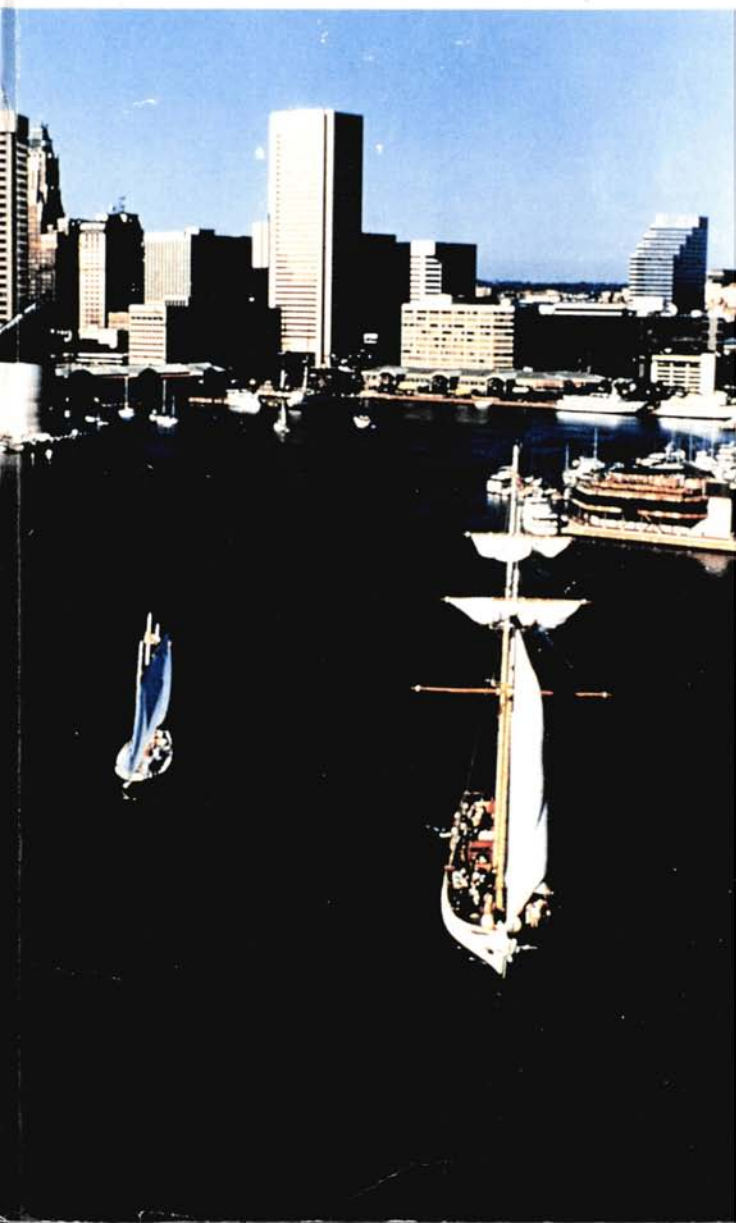
AUTO-CARTO 9

Proceedings

Ninth International Symposium on
Computer-Assisted Cartography

Baltimore, Maryland

April 2 — 7



AUTO CARTO 9
Ninth International Symposium on
Computer-Assisted Cartography
Baltimore, Maryland
April 2 — 7



Sponsored by
American Society for Photogrammetry
and Remote Sensing
American Congress on Surveying and
Mapping



©1989 by the American Society for Photogrammetry and Remote Sensing and the American Congress on Surveying and Mapping. All rights reserved. Reproductions of this volume or any parts thereof (excluding short quotations for use in the preparation of reviews and technical and scientific papers) may be made only after obtaining the specific approval of the publishers. The publishers are not responsible for any opinions or statements made in the technical papers.

COVER PHOTO

Baltimore's Inner Harbor with a tall ship in the foreground. Photo courtesy of Baltimore Area Visitor's and Convention Association.

ISBN 0-944426-55-7

Published by
American Society for Photogrammetry and Remote Sensing
and
American Congress on Surveying and Mapping
210 Little Falls Street
Falls Church, VA 22046
USA

Printed in the United States of America

FOREWORD

This volume contains papers from the Ninth International Symposium on Computer-Assisted Cartography held in Baltimore, Maryland. These Proceedings contain only those papers that were received in time for publication. Authors are listed in an alphabetical index for the convenience of the reader.

The Director and Program Committee are grateful to the authors, coauthors, and their typists who contributed their time and talents toward making this volume possible. They also wish to thank colleagues who assisted in this task.

Eric Anderson
Technical Program Coordinator

TABLE OF CONTENTS

Advanced Data Display

- A Practical and Efficient Approach to the Stereoscopic Display and Manipulation of Cartographic Objects, H. Moellering, Ohio State U. 1
- Visualization Techniques and Applications within GIS, R.A. McLaren, Know Edge, Ltd. 5
- Two-Variable Color Mapping on a Microcomputer
C.B. Dawsey III, Auburn U. 15

Data Bases - A Continuous Process

- Updating Urban Street Network Files with High Resolution Satellite Imagery, L. Li, G. Deecker, K. Yurach, and J. Seguin, Statistics Canada and Environment Canada 21
- Establishing a Corporate GIS Data Base from Multiple GIS Project Data Sets, T.R. Johnson and K.C. Siderelis, Land Resources Information Service 874

GIS Education and Training

- Education and Training in GIS: The View from ESRI, T. Burns and J. Henderson, Environmental Systems Research Institute, Inc. 31
- Geographic Information System Teaching at ITC
J. Drummond, J-C. Muller, and P. Stefanovic, ITC 38
- GIS-Related Education and Training at Siemens
H.J. Vogel, Siemens AG 47

Generalization

- Cartographic Generalization in a Digital Environment: When and How to Generalize, K.S. Shea and R.B. McMaster, The Analytic Sciences Corp. and Syracuse U. 56
- Conceptual Basis for Geographic Line Generalization,
D.M. Mark, SUNY Buffalo 68
- Data Compression and Critical Points Detection Using Normalized Symmetric Scattered Matrix, K. Thapa, Ferris State U. 78

Data Structures and Parallel Processing

- Transputer-Based Parallel Processing for GIS
Analysis: Problems and Potentialities, R.G. Healey
and G.B. Desa, U. of Edinburgh 90
- Uniform Grids: A Technique for Intersection
Detection on Serial and Parallel Machines,
W.R. Franklin, N. Chandrasekhar, M. Kankanhalli,
D. Sun, M. Zhou, and P.Y.F. Wu, Rensselaer Polytechnic
Institute 100
- A Geographic Data Model Based on HBDS Concepts:
The IGN Cartographic Data Base Model, F. Salge and
M.N. Sclafer, Institut Geographique National 110
- Demonstration of Ideas in Fully Automatic Line
Matching of Overlapping Map Data, R.J. Hintz and
M.Z. Zhao, U. of Maine 118

Data Structures and Post-Processing for Digital Terrain

- Topographic Grain Automated from Digital
Elevation Models, R.J. Pike, W. Acevedo, and D.H. Card,
USGS and NASA Ames 128
- A Spatial Low-Pass Filter Working for Triangular
Irregular Network (TIN) and Restricted by Break Lines,
Z.-T. Chen, Environmental Systems Research Institute, Inc. 138
- A Compact Terrain Model Based on Critical Topographic
Features, L.L. Scarlatos, Grumman Data Systems 146
- A Shortest Path Method for Hierarchical Terrain Models,
R. Barrera and J. Vazquez-Gomez, U. of Maine and UAM-
Atzacapotzalco 156

Topics in GIS and Automated Cartography

- Hipparchus Data Structures: Points, Lines, and Regions
in Spherical Voronoi Grid, H. Lukatela, Calgary 164
- Interactive Analytical Displays for Spatial Decision
Support Systems, M.P. Armstrong and P. Lolonis,
U. of Iowa 171
- Automated Insetting: An Expert Component Embedded in
the Census Bureau's Map Production System, A.A. Martinez,
Bureau of the Census 181
- Accessing Spatiotemporal Data in a Temporal GIS,
G. Langran, U. of Washington 191

A GIS Curriculum for Universities

Components of Model Curricula Development for GIS
in University Education, T.L. Nyerges, U. of Washington 199

Automated Names Placement

Automated Names Placement in a Non-Interactive Environment,
L.R. Ebinger and A.M. Goulette, Bureau of the Census 205

An Expert System for Dense-Map Name Placement,
J.S. Doerschler and H. Freeman, Hamilton Standard Division
and Rutgers University 215

The Use of Artificial Intelligence in the Automated Placement
of Cartographic Names, D.S. Johnson and U. Basoglu,
Intergraph Corp. 225

Rule-Based Cartographic Name Placement with Prolog,
C.B. Jones and A.C. Cook, Polytechnic of Wales 231

Digital Terrain

The Development of Digital Slope-Aspect Displays,
A.J. Kimerling and H. Moellering, Oregon State U. and
Ohio State U. 241

Conversion of Contours, B. Shmutter and Y. Doytsher,
Technion I.I.T. 245

Relative Errors Identified in USGS Gridded DEMs,
J.R. Carter, U. of Tennessee 255

GIS Design: Examining the Alternatives

The Architecture of ARC/INFO, S. Morehouse,
Environmental Systems Research Institute, Inc. 266

Algorithms

The Combinatorial Complexity of Polygon Overlay,
A. Saalfeld, Bureau of the Census 278

Pushbroom Algorithms for Calculating Distances in
Raster Grids, J.R. Eastman, Clark U. 288

Spatial Adjacency - A General Approach, C.M. Gold,
Memorial U. 298

Algorithms

- Multiscale Data Models for Spatial Analysis, with Applications to Multifractal Phenomena, L. DeCola, U. of Vermont 313

Three-dimensional GIS

- Three-Dimensional GIS for the Earth Sciences, D.R. Smith and A.R. Paradis, Dynamic Graphics, Inc. 324
- National Capital Urban Planning Project: Development of a Three-Dimensional GIS Model, L.G. Batten, USGS 336
- GIS Future: Automated Cartography or Georelational Solid Modeling?, H. Lukatela, Calgary 341

Data Capture Techniques

- Spectral/Spatial Exploitation of Digital Raster Graphic Map Products for Improved Data Extraction, T.J. Eveleigh and K.D. Potter, Autometric Inc. 348
- Cartographic Data Capture Using CAD, M.E. Hodgson, M.L. Barrett, and R.W. Plews, U. of Colorado and Hunter College, CUNY 357
- Tigris Mapper Viewed as a Digital Data Capturing Tool in Object Oriented Environment, J. Mitter, Intergraph Corp. 367
- Data Capture for the Nineties: VTRAK, R. Waters, D. Meader, and G. Reinecke, Laser-Scan Laboratories 377

Environmental Applications of GIS

- Polygon Overlay to Support Point Sample Mapping: The National Resources Inventory, D. White, K. Chan, M. Maizel, and J. Corson-Rikert, NSI Technology Support Corp. 384
- Hazardous Waste Disposal Site Selection Using Interactive GIS Technology, C. Van Zee and J.E. Lee, Ebasco Services and QC Data Collectors, Inc. 391
- Testing Large-Scale Digital Line Graphs and Digital Elevation Models in a Geographic Information System, D.R. Wolf and E.T. Slonecker, USGS and Bionetics Corp. 397

Data Structures and Spatial Query Techniques

- Quadtree Meshes, W.T. Verts and F.S. Hill, Jr.,
U. of Massachusetts 406
- Storage Methods for Fast Access to Large
Cartographic Data Collections - An Empirical
Study, A. Kleiner, U. of Zurich 416
- Solving Spatial Queries by Relational Algebra,
R. Laurini and F. Milleret, INSA-Lyon 426

Structuring Large Spatial Data Bases

- Speculations on Seamless, Scaleless, Cartographic
Data Bases, S.C. Guptill, USGS 436
- Optimal Tiling for Large Cartographic Databases,
M.F. Goodchild, UC Santa Barbara 444
- The Geographic Database - Logically Continuous
and Physically Discrete, P. Aronson, Environmental
Systems Research Institute, Inc. 452
- Planetary Modeling via Hierarchical Tessellation,
G. Dutton, Prime Computer 462

GIS Applications

- Use of the 1:2,000,000 Digital Line Graph Data in
Emergency Response, H. Walker, Lawrence Livermore
National Laboratory 472
- Use of a Geographic Information System to Evaluate
the Potential for Damage from Subsidence of Underground
Mines in Illinois, C.A. Hindman and C.G. Treworgy,
Illinois State Geological Survey 483

International Perspectives

- Digital Data: The Future for Ordnance Survey,
M. Sowton, Ordnance Survey, UK 493
- GIS, AM/FM, and Automated Cartography in Japan,
S. Kubo, Ochanomizu U. 505
- Trends of Computer-Assisted Cartography in Hungary
and Eastern Europe, P. Divenyi, Institute of Geodesy,
Cartography, and Remote Sensing 513

Quality Control Issues

- Error in Categorical Maps: Testing versus Simulation, N.R. Chrisman, U. of Washington 521
- Modeling Error for Remotely Sensed Data Input to GIS, M.F. Goodchild and M. Wang, U. of Western Ontario and U.C. at Santa Barbara 530

Spatial Relations and Data Base Models

- Concepts of Space and Spatial Language, D.M. Mark and A.U. Frank, SUNY Buffalo and U. of Maine 538
- Geographic Information: Aspects of Phenomenology and Cognition, R.J. Williams, Australian Defence Force Academy 557
- GIS Support for Micro-Macro Spatial Modeling, T.L. Nyerges, U. of Washington 567
- Context-Free Recursive-Descent Parsing of Location-Descriptive Text, M. McGranaghan, U. of Hawaii 580

Object-Oriented Approaches to GIS

- Object-Oriented Modeling in GIS: Inheritance and Propagation, M.J. Egenhofer and A.U. Frank, U. of Maine 588
- Geographic Logical Database Model Requirements, M. Feuchtwanger, U. of Calgary 599

Educational Tools for GIS

- GIST: An Object-oriented Approach to a Geographical Information System Tutor, J.F. Raper and N.P.A. Green, Birkbeck College 610
- DEMOGIS Mark 1: An ERDAS-Based GIS Tutor, D.J. Maguire, University of Leicester 620

Poster Session

- Cartographic Analysis of U.S. Topography from Digital Data, R.J. Pike and G.P. Thelin, USGS 631
- A Full Function GIS Editor, W.H. Moreland, Environmental Systems Research Institute, Inc. 641

Poster Session

- A Study of Spatial Data Management and Analysis Systems, C. Christopher and R. Galle, Jackson State U. and Stennis Space Center 648
- Sliding Tolerance 3-D Point Reduction for Globograms, S. Prashker, Carleton U. 655
- A Reactive Data Structure for Geographic Information Systems, P. van Oosterom, U. of Leiden 665

Challenges for the Future

- Challenges Ahead for the Mapping Profession, J.C. Müller, ITC 675
- An On-line, Secure and Infinitely Flexible Data Base System for the National Population Census, D.W. Rhind, E. Hayes-Hall, H.M. Mounsey, and S. Openshaw, Birkbeck College and The University of Newcastle 684

Automated Mapping Applications

- A Cartographic Extract of the TIGER File: Implications for Mapping Applications, A. Bishton, Bureau of the Census 697
- A Versatile Mapping System for the USGS 1:100,000 DLGs, D.J. Cowen and T.R. White, U. of South Carolina 705
- New York State's Digital County Mapping Program, T.W. Koch, NY DOT 715
- Vector-Based Computer Graphics in Automated Map Compilation, C.F. Scheepers, CACDS 724

Standards and Their Use

- First UNIX, then UGIX, D.W. Rhind, J.F. Raper, and N.P.A. Green, Birkbeck College 735
- The South African Standard for the Exchange of Digital Geo-Referenced Information, A.K. Cooper, CACDS 745
- The Telecommunication of Map and Chart Data, T. Evangelatos, Z. Jiwani, D. McKellar, and C.D. O'Brien, Canadian Hydrographic Service, OMNR, DND, IDON Corp. 754

GIS: Directions for the Future

The ESRC's Regional Research Laboratories: An Alternative Approach to the NCGIA?, J.W. Shepherd, I. Masser, M. Blakemore, and D.W. Rhind, Birkbeck College, U. of Sheffield, and U. of Durham 764

The Institutional Context of GIS: A Model for Development, P.F. Fisher and M.N. DeMers, Kent State U. and Ohio State U. 775

GIS Performance

The Power of Symbology in the GIS World, M.E. Gentles, Synercom Technology, Inc. 781

On the Design of Geographic Information System Procedures, J.A. Guevara, Environmental Systems Research Institute, Inc. 789

Performance Testing of Gridcell-Based GIS, S.E. Amundson, U. of Hawaii at Hilo 798

Use Error: The Neglected Error Component, K. Beard, U. of Maine 808

System Design, Integration, and Application

Extending Entity/Relationship Formalism for Spatial Information Systems, Y. Bedard and F. Paquette, Laval U. 818

A Fully Integrated Geographic Information System, J.R. Herring, Intergraph Corp. 828

AM/FM and GIS

Spatial Tools for the Administration of Major Institutions, J.M. Young, Program Administration Group 838

Displays for Spatial Data

Incorporating the Laborde Projection into an Existing Cartographic Software Package, P.H. Laskowski, Intergraph Corp. 850

IBM PC Animation - Crude but Effective, W.T. Verts, U. of Massachusetts 858

CAD: A Viable Alternative for Limited Cartographic and GIS Applications, R.C. Anderson and L.D. Carmack, Jr. U.S. Military Academy 867

Author Index

W. Acevedo	128
S.E. Amundson	798
R.C. Anderson	867
M.P. Armstrong	171
P. Aronson	452
R. Barrera	156
M.L. Barrett	357
U. Basoglu	225
L.G. Batten	336
K. Beard	808
Y. Bedard	818
A. Bishton	697
M. Blakemore	764
T. Burns	31
D.H. Card	128
L.D. Carmack, Jr.	867
J.R. Carter	255
K. Chan	384
N. Chandrasekhar	100
Z.-T. Chen	138
N.R. Chrisman	521
C. Christopher	648
A.C. Cook	231
A.K. Cooper	745
J. Corson-Rikert	384
D.J. Cowen	705

C.B. Dawsey III	15
L. DeCola	313
G. Deecker	21
M.N. DeMers	775
G.B. Desa	90
P. Divenyi	513
J.S. Doerschler	215
Y. Doytsher	245
J. Drummond	38
G. Dutton	462
J.R. Eastman	288
L.R. Ebinger	205
M.J. Egenhofer	588
T. Evangelatos	754
T.J. Eveleigh	348
M. Feuchtwanger	599
P.F. Fisher	775
A.U. Frank	538, 588
W.R. Franklin	100
H. Freeman	215
R. Galle	648
M.E. Gentles	781
C.M. Gold	298
M.F. Goodchild	444, 530
A.M. Goulette	205
N.P.A. Green	610, 735
J.A. Guevara	789

S.C. Guptill	436
E. Hayes-Hall	684
R.G. Healey	90
J. Henderson	31
J.R. Herring	828
F.S. Hill, Jr.	406
C.A. Hindman	483
R.J. Hintz	118
M.E. Hodgson	357
Z. Jiwani	754
D.S. Johnson	225
T.R. Johnson	874
C.B. Jones	231
M. Kankanhalli	100
A.J. Kimerling	241
A. Kleiner	416
T.W. Koch	715
S. Kubo	505
G. Langran	191
P.H. Laskowski	850
R. Laurini	426
J.E. Lee	391
L. Li	21
P. Lolonis	171
H. Lukatela	164, 341
D.J. Maguire	620
M. Maizel	384

D.M. Mark	68, 538
A.A. Martinez	181
I. Masser	764
M. McGranaghan	580
D. McKellar	754
R.A. McLaren	5
R.B. McMaster	56
D. Meader	377
F. Milleret	426
J. Mitter	367
H. Moellering	1, 241
S. Morehouse	266
W.H. Moreland	641
H.M. Mounsey	684
J.C. Muller	38, 675
T.L. Nyerges	199, 567
C.D. O'Brien	754
P. van Oosterom	665
S. Openshaw	684
F. Paquette	818
A.R. Paradis	324
R.J. Pike	128, 631
R.W. Plews	357
K.D. Potter	348
S. Prashker	655
J.F. Raper	610, 735
G. Reinecke	377

D.W. Rhind	684, 735, 764
A. Saalfeld	278
F. Salge	110
L.L. Scarlatos	146
C.F. Scheepers	724
M.N. Sciafer	110
J. Seguin	21
K.S. Shea	56
J.W. Shepherd	764
B. Shmutter	245
K.C. Siderelis	874
E.T. Slonecker	397
D.R. Smith	324
M. Sowton	493
P. Stefanovic	38
D. Sun	100
K. Thapa	78
G.P. Thelin	631
C.G. Treworgy	483
C. Van Zee	391
J. Vazquez-Gomez	156
W.T. Verts	406, 858
H.J. Vogel	47
H. Walker	472
M. Wang	530
R. Waters	377
D. White	384

T.R. White	705
R.J. Williams	557
D.R. Wolf	397
P.Y.F. Wu	100
J.M. Young	838
K. Yurach	21
M.Z. Zhao	118
M. Zhou	100

**A PRACTICAL AND EFFICIENT APPROACH TO THE
STEREOSCOPIC DISPLAY AND MANIPULATION
OF CARTOGRAPHIC OBJECTS**

by

Prof. Harold Moellering
Department of Geography
Ohio State University
Columbus, Ohio
U.S.A. 43210
Bitnet: TS0215@OHSTVMA

ABSTRACT

Until now the direct stereoscopic display of computer generated cartographic objects produced in real time has been either very expensive or very difficult. This paper presents an approach that is both more efficient than earlier methods and more practical. Several examples of cartographic surfaces will be shown where the stereoscopic vision aspect of the display can be used to more efficiently show the true character of the surfaces involved.

INTRODUCTION

For many decades cartographers have been faced with the challenge of showing a three dimensional cartographic surface on a two dimensional medium. This includes both terrain and thematic cartographic data. In the early decades the medium was the paper map whereas in more recent years graphic CRT displays showing virtual maps have largely taken over this task. Common approaches to this problem are to use isarithmic representations, profiles, meshes, or choroplethic representations for such data. Approaches known as analytical hill shading are also used. Horn (1982) provides a review of these methods used in cartography as well as some of the shading approaches used in computer graphics. However, the challenge of rendering a cartographic surface as a true three dimensional surface still remains. This paper describes an approach to directly creating true three dimensional color cartographic displays through the use of stereoscopic CRT displays.

EARLY APPROACHES TO STEREOSCOPIC DISPLAYS

One of the earliest approaches to the direct stereoscopic display of cartographic surfaces was developed with anaglyphs. Here two images of the same surface, usually of single or double profiles, were printed in red and blue ink on paper and slightly offset from each other. This twin image was then viewed through glasses that had a red lens for one eye and a blue lens for the other. This

produced a true three dimensional effect, but correct colors could not be shown. A more mechanical traditional approach has been taken by photogrammetry where two air photos which overlap each other have been taken from slightly different angles, called parallax. These two photos can then be viewed with a device called a stereoscope to use the parallax to achieve a true three dimensional display of the terrain surface. Over the last several decades such mechanical photogrammetric devices have greatly increased in sophistication to increase the fidelity of the image. However, such devices still are limited to visible terrain and are not useful for viewing thematic surfaces such as population or temperature. A more recent innovation is the development of dual vision goggles which contain two separate television displays, one for each eye. Such devices are being used experimentally in a number of areas, but so far are limited to such experimental applications. Many of these dual vision devices significantly strain the eyes and hence are not practical for regular scientific work.

A MORE PRACTICAL AND EFFICIENT APPROACH

An approach that is far more practical and efficient for the average scientific user is a solution that has been made available by the Tektronix company recently in its 4200 series terminals and 4300 series workstations. This kind of stereoscopic display is generated from a single CRT screen workstation that creates the left and right images essentially simultaneously by alternating the images at the very fast rate of 120 Hertz noninterlaced. Therefore each image, left and right, is refreshed individually at a 60 Hertz rate. To the viewer this rate is perceived as continuous. These images are displayed through a liquid crystal polarizer that can be rapidly changed at the same 120 Hertz rate as the refresh cycle in the color CRT. The output from the polarizer is colored light that has been clockwise polarized for one image and counterclockwise polarized for the other. Radial polarization has been used here because it provides a much wider field of view than would more conventional linear polarization that is found in more conventional applications. The viewer then uses special radially polarized glasses to view the image stereoscopically. Since each lens of the glasses is radially polarized in opposite directions, the viewer sees a different image with each eye. Hence true 3-D stereoscopic display is achieved in a simple and efficient manner.

This approach can be explained in a more technical fashion as shown in Figure 1. The U,V coordinate display windows contains the point in their center called VRP, the View Reference Point. This VRP is then defined as the center of the viewing space. Usually there is only one Viewing Plane Normal, VPN, but since this is a stereoscopic display, two viewing plane normals are used, VPN-R and VPN-L, one for the left and one for the right portion of the stereoscopic image. Associated with each of the viewing plane normals is a viewing window, UV Window-R and

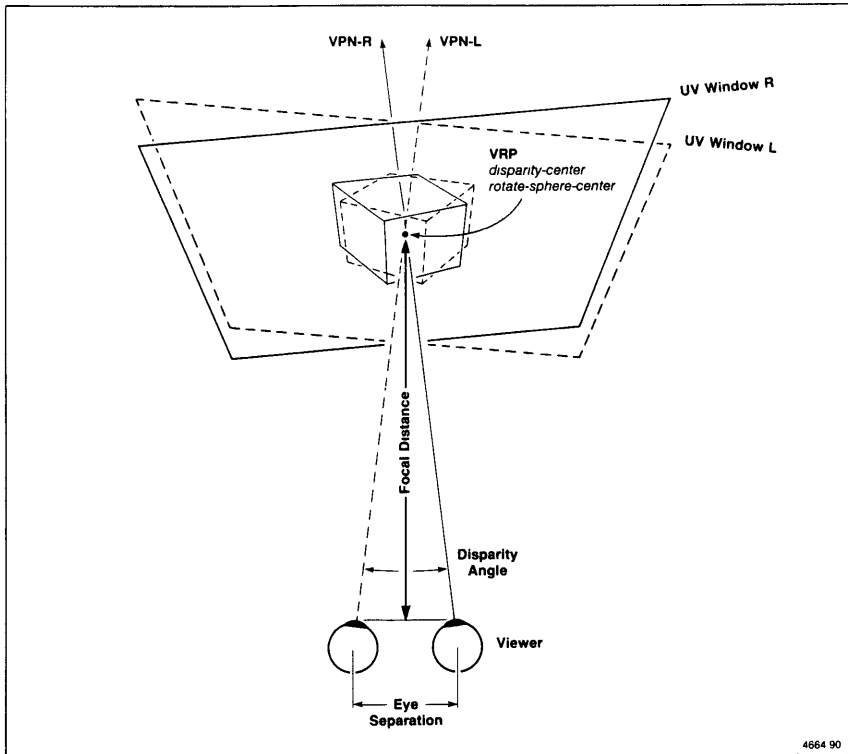


Figure 1. Stereoscopic Viewing Using Left and Right Images
(Figure © Copyright Tektronix Inc. 1988)

UV Window-L. Each of these displays one component of the Left/Right stereo image. The angle between VPN-R and VPN-L is referred to by Tektronix as the Disparity Angle which is the angle of parallax between the Left and Right portions of the stereo display. The size of this angle is influenced by the size and color of the object(s) being displayed, as well as the stereoscopic perception characteristics of the individual viewer. The default is four degrees. With this kind of display and subsequent polarization, the right eye receives only the Right image while the left eye only receives the Left image. Therefore efficient stereoscopic vision is achieved.

Using such an approach to stereoscopic viewing produces a display that is both practical and efficient, and adds another dimension to the manipulation and display of virtual maps in interactive real time. It is practical because the stereoscopic image looks very real. The CRT display preserves the proper colors numerically assigned in the computer program driving the display. Use of the

stereoscopic display is very effective because the radial polarization provides a wide angle of view such that several people can view the image at the same time. The practicality of the display is further appreciated because all of the stereoscopic image generation is done internally in the electronics of the CRT hardware and does not require any special software development by the user, although the disparity angle may have to be adjusted for individual viewers. Therefore, any computer display program that runs properly on the lower level Tektronix equipment will probably run on the stereoscopic display equipment with no modification. This stereoscopic display is also more efficient because it can be used in interactive real time. Therefore all of the interactive design manipulation strategies that one would usually associate with 3-D virtual maps (Moellering, 1977) can be enhanced with this kind of true three dimensional stereoscopic display.

CONCLUSION

As is clear from the discussion above, this new approach to the digital stereoscopic display of cartographic data using the Tektronix radial polarization technique is very effective for cartographic applications. It is convenient to use and easy for the programmer in that no new special software is necessary to implement this approach. At the same time this approach does not suffer the disadvantages of earlier attempts.

ACKNOWLEDGEMENTS

This research work is supported by a grant from the NASA Center for Real-Time Satellite Mapping at Ohio State University. Tektronix is a commercial partner in this project, and gave permission to use the figure in this paper. Mr. Peter Dotzauer implemented the software for the system discussed. Prof. Jon Kimerling of Oregon State University provided valuable comments during this work. The U.S. Geological Survey provided the data used in the examples.

REFERENCES

- Horn, B.K.P., 1982, Hill Shading and the Reflectance Map, Geoprocessing, 2: 65-146.
- Moellering, H., 1977, Interactive Cartographic Design, Proceedings of the American Congress on Surveying and Mapping, 37th Annual Meeting, 516 - 530.
- Tektronix Inc., 1988, 2-D/3-D Graphics Programmer, Volume 2, Beaverton, Oregon: Tektronix Inc.

VISUALISATION TECHNIQUES AND APPLICATIONS WITHIN GIS

Robin A. McLaren
Know Edge Ltd
33 Lockharton Avenue
Edinburgh EH14 1AY Scotland

ABSTRACT

The recent advances in computer graphics rendering techniques coupled with the availability of digital terrain model information have opened up new possibilities in viewing and analysing spatial information within a GIS environment. Terrain and landscape visualisation techniques ranging from simple wire-frame models through to photorealistic rendering approaches such as ray tracing and radiosity are reviewed. The techniques are then placed within the context of a variety of GIS applications including: spatial analysis, urban planning, architectural design, cartography, highway and traffic engineering, and environmental impact assessment.

INTRODUCTION

Increasing emphasis has been directed recently towards the development of computer based display techniques which combine a high degree of image realism with a high level of symbolism. Where such techniques are used to describe the 3-D shape of the earth's surface and man made or other 'cultural' information, the process is referred to as digital terrain and landscape visualisation.

The input data for such a process will normally consist of a digital terrain model (DTM) to define the geometric shape of the earth's surface, together with further geometric and descriptive data to define landscape features. For applications at small scales, this landscape information may be polygonal land use data, while at larger scales it may include explicit 3-D geometric descriptions of individual features or blocks of features. Although the photogrammetric acquisition of DTM data and its subsequent management within a GIS environment is well established, to date, much less attention has been directed towards the acquisition and management of 3-D geometric and visual descriptions of significant objects on the terrain. Normally, the 3-D co-ordinates of features such as buildings are observed during photogrammetric measurement, but in many cases the Z co-ordinate values are deemed superfluous and do not form part of the recorded dataset. This is primarily due to the inadequacy of GIS to handle data structures associated with volumes. Hence the more sophisticated visualisation techniques are currently being performed on CAD/CAM oriented systems.

While there is no shortage of techniques for general visualisation purposes, the characteristics of the earth's surface can significantly limit the applicability of many of the more general purpose techniques. Some of the more

important differences between terrain and landscape visualisation and other forms of visualisation, especially those found in the advertising and entertainment industries, are that:

- (a) natural phenomena are inherently more complex than man made objects and thus more difficult to model;
- (b) the earth's surface is not geometric in character and cannot be modelled effectively by using higher order primitives such as those used in solid modelling CAD / CAM applications;
- (c) the terrain and landscape model dataset sizes are considerably larger than the datasets in many other forms of visualisation;
- (d) there are generally higher constraints on geometric accuracy than in many other applications;
- (e) the scenes are not spatially compact and therefore the modelling may involve multiple levels of detail based on the object/viewpoint relationship;
- (f) the optical model is more complex due to the effects of atmospheric refraction and earth curvature which are encountered in extensive datasets.

These inherent complexities of natural phenomena have inhibited the wider use of available computer graphic tools. However, recent advances in the sophistication of the rendering algorithms, the lowering in cost of high performance imaging systems, the emergence of PC based rendering platforms, and the arrival of standards such as RenderMan will help these visualisation techniques to filter down to low-cost systems, adding another valuable technical as well as marketing component to the GIS toolkit.

METHODS OF RENDERING 3-D TERRAIN AND LANDSCAPE MODELS

The degree of realism which can be achieved in the visualisation process is dependent upon several factors including the nature of the application, the objective of the visualisation, the capabilities of the available software and hardware and the amount of detail recorded in the model of the scene. At one end of the realism spectrum are relatively simple, highly abstract, static, monochrome, wireframe models while at the opposite end are sophisticated, highly realistic, dynamic full colour images.

A number of alternative strategies exist for the transformation and display, or 'rendering' of 3-D terrain and landscape models onto a 2-D raster scan display. The first, and computationally the simplest, involves the use of a video digitiser to 'frame grab' a photographic image. The position of new features may then be added to the model. The second, and computationally the most complex approach, is to mathematically define all features within the model. The basic elements of this approach are discussed in section 2.1. In some cases a hybrid approach may be adopted where photographic images are used to model complex natural phenomena and are combined with terrain and landscape features which are modelled by computer

graphics. This is described in section 3.1.

Review of the Rendering Process

Having assembled the model of the terrain and associated landscape features contained within the required scene, the scene may then be rendered onto a suitable display system. The rendering process, however, requires a number of other parameters to be defined (Figure 1) including:

- (a) the viewing position and direction of view of the observer;
- (b) a lighting model to describe the illumination conditions;
- (c) a series of "conditional modifiers", parameters which describe the viewing condition of the landscape objects (under wet conditions, for example, the surface characteristics of objects are quite different from dry conditions);
- (d) a set of "environmental modifiers", parameters which describe atmospheric conditions and may model effects such as haze; and
- (e) a sky and cloud model representing the prevailing conditions.

All, or part, of the information may then be used by the scene rendering process to generate a two dimensional array of intensities or pixel values that will be displayed on the raster display device. The complexity of the rendering process is directly dependent upon the degree of image realism required by the user and an overview of the rendering process is shown in Figure 1.

Geometric Transformations The 3-D terrain and landscape information is normally mapped into 2-D space by a perspective projection, where the size of an object in the image is scaled inversely as its distance from the viewer. For site specific applications where the geometric fidelity of the rendered scene is of vital importance, for example the creation of a photomontage product in visual impact assessment, it may also be necessary to incorporate both earth curvature and atmospheric refraction corrections into the viewing model.

Depth Cueing When a 3-D scene is rendered into 2-D space with any level of abstraction, the result is often ambiguous. To compensate for this loss of inherent 3-D information, a number of techniques have been developed to increase the 3-D interpretability of the scene using depth cueing techniques that attempt to match the perceived computer generated image to our "natural" visual cue models.

Firstly, depth cues are inherent in the perspective projection used to create the 2-D image and are emphasised when the scene contains parallel lines. This very effective visual cueing is highlighted when comparing the projection of a triangular DTM with its equivalent square grid derivative, in their wire frame forms. The depth cueing is pronounced in the square grid form due to the

Hidden Surface Removal Hidden surface removal techniques are employed to remove the edges and surfaces that are obscured by other visible surfaces. The implementation of the technique of hidden surface removal is computationally expensive, especially for complex landscape scenes, where the rendering process can involve hundreds of thousands of surfaces. Therefore the challenge has encouraged a wide variation of algorithms. In this application area one of the most popular algorithms used is the Z-buffer or refresh buffer image space algorithm.

Anti-Aliasing Many computer graphics images displayed on raster display devices exhibit disturbing image defects such as jaggling of straight lines, distortion of very small or distant objects and the creation of inconsistencies in areas of complicated detail. These distortions are caused by improper sampling of the original image and are called aliasing artefacts. Techniques known as anti-aliasing, which have their roots in sampling theory, have been developed to reduce their influence (Crow, 1977).

Shading The next step towards the goal of realism is the shading of visible surfaces within the scene. The appearance of a surface is dependent upon the type of light source(s) illuminating the object, the condition of the intervening atmosphere, the surface properties including colour, reflectance and texture, and the position and orientation of the surface relative to the light sources, other surfaces and the viewer. The objective of the shading stage is to evaluate the illumination of the surfaces within the scene from the viewer's position.

The effectiveness of the shading algorithm is related to the complexity of the model of the light sources. Natural lighting models, in the case of landscapes under daylight conditions, are normally simplified by assuming that there is only a single parallel light source i.e. the Sun. Refinements to this light model have been developed by Nishita and Nakamae (1986), in which the lighting model is considered to be a hemisphere with a large radius that acts as a source of diffuse light with non-uniform intensity, thus simulating the varying intensity of sky lighting.

There are two types of light sources apparent in the environment: ambient and direct. Direct light is light striking a surface directly from its source without any intermediate reflection or refraction while Ambient light is light reaching a surface from multiple reflections from other surfaces and the sky. A number of levels of sophistication in the modelling of light are available. The more complex modelling of ambient lighting has produced enhanced realism using ray tracing techniques (Whitted, 1980) to model the contribution from specular inter-reflections and transmitted rays, and radiosity techniques (Goral et al, 1984) to account for complex diffuse inter-reflections.

When shading DTMs, the intensity of colour is calculated at each of the vertices of the polygonal mesh and then expanded using interpolation techniques to encompass the surfaces. This is achieved by interpolating using the Gouraud approach (Gouraud, 1971), or alternatively the computationally more expensive approach developed by Phong (1975).

Shadows Shadows are an essential scene component in conveying reality in a computer graphics image. A scene that appears "flat" suddenly comes to life when shadows are included in the scene, allowing the comprehension of spatial relationships amongst objects. A variety of shadow algorithms have been developed and can be categorised into five groups : Z-buffer, area subdivision, shadow volumes, pre-processing and ray tracing.

Surface Texture Detail Natural landscape scenes are characterised by features with a wide variety of complex textures. Computer graphics visualisations of landscapes can only achieve an acceptable level of realism if they can simulate these intricate textures. The "flat" shading algorithms, described in the previous section, do not meet this requirement directly since they produce very smooth and uniform surfaces when applied to planar or bicubic surfaces. Therefore the shading approach must be supplemented by other techniques to either directly model or approximate the natural textures.

The explicit modelling approach involves creating a more detailed polygonal and colour model of the landscape and surface features to enable a higher level of detail and texture to be visualised. For landscape visualisation, explicit modelling has so far proved impractical due to the size and intricacy of the model that would have to be created to reflect the required level of detail.

Texture mapping provides the illusion of texture complexity at a reasonable computational cost. The approach refined by Blinn and Newell, 1976 is essentially a method of "wallpapering" existing polygons with a user defined texture map. This texture map can for example, represent frame grabbed images of natural textures. A modification of the texture mapping technique is to utilise satellite remote sensing imagery to "clothe" the terrain model with natural textures. This is one of the more common methods in small scale GIS applications.

A further approach for texturing terrain models involves the use of fractal surfaces (Mandelbrot, 1982) where simple models of the terrain are defined using quadrilaterals or triangles, which are subsequently recursively subdivided to produce more detailed terrain models.

Atmospheric Attenuation Due to atmospheric moisture content, objects undergo an exponential decay of contrast with respect to distance from the viewpoint. The decay converges to the sky luminance at infinity. This reduction rate is dependent upon the season, weather conditions,

level of air pollution and time. The result is a hazing effect.

APPLICATIONS

The use of visualisation techniques for both military and civilian applications is currently an area of significant growth. Some of the more prominent GIS applications are described in the following sections.

Landscape Planning - Visual Impact Analysis

Growing public awareness of environmental issues has been recently strengthened by the European Community's Directive on the "Assessment of the Effects of Certain Public and Private Projects on the Environment". This Directive will force certain proposed changes to the landscape to be publicly assessed for environmental impact. A component of this environmental audit is a statement on the visual intrusion of proposed landscape changes. Consequently, projects such as road construction, transmission line routing and open cast mining as well as more dynamic phenomena such as forestry will need to be visually judged.

Traditionally, landscape visualisation techniques have involved the building of physical models or the creation of artist's impressions. However, these are time consuming to create, and are inherently inaccurate and inflexible once created. In order to more accurately quantify the level of visual intrusion, computer graphic modelling and visualisation techniques are increasingly being used in the planning and design of landscape projects. These new approaches allow more accurate visualisations and more analytical assessments of visual intrusion to be determined. Due to the flexibility of the approach, many more proposed designs can be evaluated, resulting in a more refined design solution. Turnbull et al (1986) pioneered the development of a Computer Aided Visual Impact Analysis system (CAVIA) that has been used, for example, to provide evidence at public inquiries related to electricity transmission line routing through environmentally sensitive landscapes. Projects are typically performed at the sub-regional level with areas up to 40 x 40 km being analysed. The approach uses DTMs, landscape features and proposed design objects to produce an estimate of the visual intrusion. This visual intrusion toolkit includes intervisibility analysis to produce levels of visual impact, dead ground analysis, identification of the portions of the landscape forming a back-cloth for the design object, situations where the design object appears above the landscape horizon and the identification of optimal locations for vegetation screen placement.

One of the visualisation techniques used by CAVIA and other landscape planning systems is Photomontaging. A Photomontage is a physical or image composite of photographs of the existing landscape with a registered computer generated image of the proposed design object(s).

In this approach only the proposed design objects have to be rendered, avoiding the rendering of the intricate terrain and landscape detail. However, to achieve total image fidelity, the computer rendered portion of the image must effectively "merge" with the photographic image. Therefore, the atmospheric and distancing effects apparent in the photographic image must be inherited by the computer generated image. Nakamae, (1986) has developed techniques for merging image components to compensate for fog and aliasing effects. Future solutions to this problem will use a hybrid approach to rendering a photomontage image, incorporating ray tracing, frame grabbed images and textures, image processing and pixel painting techniques.

Road/Traffic Engineering

Visualisation has found a number of interesting applications in the field of road design. Many road engineering design systems are now offering visualisation capabilities. These form an integral part of the design process and allow the design to be subjectively assessed and refined for safety and visual intrusion in the context of its environment. The Transport and Road Research Laboratory of the UK have developed a system to model and visualise road designs. Applications of the system include;

(i) Road Lighting Scheme Design. In complex road designs or under environmental constraints, the design of an efficient road lighting scheme can be enigmatic. This is the perfect design environment for the application of visualisation techniques where the designer can directly examine the results of his design under a variety of atmospheric conditions.

(ii) Road Safety. Visualisation tends to imply aesthetic appearance. However, in this application, visualisation is concerned with the perceptual problems encountered by road users. Factors contributing to potential perceptual problems could be line of sight difficulties, incorrectly positioned street furniture or poor lane markings. In accident prone sites, the system can be used to identify possible contributing factors and to evaluate solutions.

(iii) Street Furniture Design and Placement. New designs for road signs and street furniture and their optimal positioning can be evaluated. This application has been taken one step further by the West German car manufacturer, Daimler-Benz, who have added dynamics and created a car simulator. The driver can experience driving a range of cars under a variety of driving conditions.

Architecture and Urban Design

In recent years urban renewal has become an activity increasingly exposed to and controlled by public and Royal opinion. Architects, in an attempt to alleviate public fears of a continuation of the "Kleenex Box" era, have turned to computer generated images to convince the public of the merits of their proposed building designs. Computer generated visualisations have become a fashionable marketing tool.

Although the architectural industry was one of the first application areas where Computer Aided Design (CAD) techniques were applied, it is only recently that tools for creating high quality visualisations of the resulting building designs have been made available. This capability is a natural extension of the CAD process and many CAD system vendors are now supplying this capability as an integral part of their system or as an interface to foreign visualisation packages.

Typically, architectural visualisations are not just isolated previews of the proposed building, but also include the contextual surroundings to allow appraisal of its applicability to the existing character of its urban environment. This usually involves the creation of a three dimensional model of the terrain, streets, street furniture and buildings in the immediate vicinity of the site. This approach was recently followed by Arup Associates in their submission for the development of Paternoster Square in London. The computer model of the development was created on a McDonnell-Douglas IGS system and supplemented with the surrounding urban details through photogrammetric measurement and direct input from the Ordnance Survey's digital map series.

CONCLUSIONS

Computer generated visualisations of digital terrain and landscape scenes are now widely accepted in many application areas as efficient technical analysis, design and marketing tools. Visualisation techniques have released the world from its traditional two dimensional approaches to display and in so doing, have highlighted the three dimensional deficiencies in our sources of data in terms of availability and accuracy. Indeed it is the lack of data that is currently inhibiting the wider application of many of these techniques.

Despite realism being a distant target, it acts as a convenient measure of our techniques and understanding and will continue to be relentlessly pursued to our continuing benefit. Continued increases in processing power through highly parallel architectures and customised VLSI will encourage the pursuit of the ultimate solution through the simulation of the phenomena based on the laws of physics. The present techniques of approximating or faking will be displaced by progressive refinements of the simulation model. This approach has been endorsed by the McCormick et al (1987) initiative on Visualisation in Scientific Computing.

In the GIS environment, visualisation techniques are recognised as an invaluable system component, aiding in the interpretation of spatially related phenomena and complex data analyses that takes the GIS a step beyond two dimensional polygonal overlay analyses. Many of the GIS vendors are including this capability in their systems to help cope in our understanding of the "fire hose" of data being produced by contemporary sources such as satellites.

REFERENCES

- BLINN J.F. and NEWELL M.E 1976. Texture and Reflection in Computer Generated Images, Communications of the ACM, October, Vol 19, Number 10, 542-547.
- CROW F.C. 1977. The Aliasing Problem in Computer Generated Shaded Images, Communications of the ACM, November, Vol 20, Number 11, 799-805.
- GORAL C.M., TORRANCE K.E., GREENBERG D.P. and BATTAILE B. 1984. Modeling the Interaction of Light Between Diffuse Surfaces, ACM Computer Graphics, July, Vol. 18, Number 3, 213-222.
- GOURAUD H. 1971. Continuous Shading of Curved Surfaces, IEEE Transaction on Computers, June, Vol C-20, Number 6, 623-629.
- KENNIE T.J.M. and McLAREN R.A. 1988. Modelling for Digital Terrain and Landscape Visualisation. Photogrammetric Record, 12(72): 711-745.
- MANDELBROT B. 1982. The Fractal Geometry of Nature, W.H. Freeman, San Francisco. 468 pages.
- MCCORMICK B.H., DEFANTI T.A. and BROWN M.D., 1987. Visualisation in scientific computing - a synopsis. IEEE Computer Graphics and Applications, 7(7): 61-70.
- NAKAMAE A., HARADA K., ISHIZAKI T. and NISHITA T., 1986. A montage method: the overlaying of the computer generated images onto a background photograph. ACM Computer Graphics, 20(4): 207 - 214.
- NISHITA T. and NAKAMAE E. 1986. Continuous Tone Representation of Three-Dimensional Objects Illuminated by Sky Light, ACM Computer Graphics, July, Vol 20, Number 4, 125-132.
- TURNBULL W.M., MAVER T.W. and GOURLAY I., 1986. Visual impact analysis: a case study of a computer based system. Proceedings Auto Carto London, 1: 197-206.
- WHITESIDE A. ELLIS M., and HASKELL B. 1987. Digital Generation of Stereoscopic Perspective Scenes. SPIE Conference on True Three Dimensional Imaging Techniques and Display Technologies, Vol. 761, 7pp.
- WHITTED T. 1980. An Improved Illumination Model for Shaded Display, Communications of the ACM, June, Vol 23, Number 6, 343-349.

TWO-VARIABLE COLOR MAPPING ON A MICROCOMPUTER

Cyrus B. Dawsey III
Auburn University
Department of Geography
Auburn, AL. 36849

BIOGRAPHICAL SKETCH

Cyrus B. Dawsey received a PhD from the University of Florida and has spent his professional career at Auburn University where he is an Associate Professor of Geography. His research interests have included economic spatial patterns, migration, allocation models, cartography, and computer applications. Dr. Dawsey has maintained a regional focus in Latin America and Brazil where he spent much of his early life.

ABSTRACT

Though two-variable choropleth maps are useful for analyzing the direction, intensity and spatial pattern of a correlation, they are not frequently used because costs associated with conventional production are made high by color layout requirements and by the limited demand for maps of any particular combination of variables. Microcomputers with color output capability make such maps available to a potentially wider audience. A properly programmed system can create an almost unlimited number of unique two-variable maps at little expense beyond the initial investments. This report describes the development and illustrates the results of two-variable choropleth mapping on the Amiga microcomputer.

INTRODUCTION

A two-variable choropleth map can be a useful device for representing the direction, intensity, and spatial patterning of a correlation between two sets of data. If enumeration area observations for variables 1 and 2 are grouped into n and m classes, a rectangular legend with $n \times m$ cells displays all category combinations for the variables arrayed along the x and y axes. Most two-variable maps use color to distinguish patterns, so an increase in the perception of a selected hue usually indicates increasing values on the legend axes. Two diagonally opposite corners of the legend, each corresponding to the highest values for one variable and lowest for the other, present cells that are completely saturated in one of the colors. The other two corners are neutrally toned. Low values for both variables are commonly shown in black and high values for both variables in white.

Direction and intensity of a correlation is given by the position of the prevalent map hues relative to the legend diagonals. If most of the map appears in neutral shades from white to black the variables are positively associated, but the relationship is negative if the majority of the areas are vividly saturated in the two

colors. The strength of the correlation is roughly indicated by the percentage of the total area mapped in tones along one of the diagonals.

An advantage of two-variable maps is that they provide a distributional representation of the correlation information. Spatial variations in the direction, intensity, and residuals of a relationship between paired variables are more readable on a map than in a table of coefficients. For example, areas of low and high value for each separate variable as well as areas of low and high joint value are immediately identifiable on a single map.

Though often discussed as a useful technique, two-variable mapping is not widely used (Meyer,1975;Olson,1981). Cost is an important factor. Scholarly publishers usually operate on budgets that preclude extensive use of color, while the better heeled cartography companies print maps for a general readership that is seldom interested in multivariable representation. This limited interest may in turn be a function of the scarcity of two-variable maps. A certain familiarity and history of exposure may be necessary before an average person is able to fully comprehend the patterns and relationships depicted on such maps.

Another factor which may limit the availability of two-variable maps is the inherently vast potential for thematic selection. By general consensus, single-variable maps should depict standard items such as landforms, population, or transportation, so heavy production investment is justified by the guaranteed mass demand. No such agreement exists regarding the selection of combinations of variables to be shown on two-variable maps. The relatively small audience for any given map and the high cost of a full color layout have limited most two-variable cartography to demonstration issues. Full use of this type of map for data analysis is relatively rare.

COMPUTER APPLICATIONS

Computer mapping can overcome some of the conditions that limit the use of two-variable maps. Today, relatively inexpensive systems are capable of displaying multi-hued graphics, and they can be programmed to produce choropleth maps. Once the hardware and software have been purchased, a virtually infinite number of multi-colored maps representing varied themes and areas can be generated at no extra cost. High expense per unique map, the drawback of conventional production, can be avoided. Though physical output (hardcopy) may involve money or time outlays, direct analyses of two-variable maps is available to anyone with access to a properly programmed computer connected to a color CRT.

The remainder of this paper describes some aspects of the development of a two-variable choropleth mapping program for the Amiga computer. The introduction of the Amiga 1000 in 1985 generated much excitement in the microcomputer world (Williams,1985;Anderson,1985). Reviewers described

it as a landmark machine with capabilities unavailable in any other comparably priced (\$1000 - \$2000) system. Its sound chip produced music in stereo and talked in English and other languages; its Motorola 68000 processor provided true multitasking by running applications independently of each other; and its architecture was open for easy expansion. The most attractive feature of the Amiga, however, was its ability to produce colorful graphics. Dealers fondly displayed bouncing balls and multi-hued robots to demonstrate the selection palette of over 4000 colors on a screen resolved at up to 640 x 400 pixels. Furthermore, because the hardware was designed to minimize the time required for generating graphic output, convincing animation effects were possible. While written line commands provide complete access to DOS, a mouse controlled environment of icons and windows was the primary user interface.

An effective choropleth mapping system must bring together 3 basic elements; basemap information, the data values to be mapped, and instructions on how to produce the map. Three program modules were designed to accomplish this on the Amiga. The software allowed for the following. Basemap files were created with the mouse which is standard equipment on the Amiga, and the digitizing source consisted of a map outlined on a transparent sheet affixed over the monitor. The cursor was positioned on the screen at successive map boundary intersections or border direction change points, and each was identified as an x-y coordinate of a line segment endpoint by clicking the mouse button. Keyboard editing controls were incorporated, and, once completed, the basemap file was stored to disk.

Development of the second and perhaps most important element, a data base of the variables to be mapped, was straightforward. The number of enumeration units, their names, and the associated data values were stored in a sequential file. Data input to the file was facilitated by a routine which displayed an outline map successively highlighting each area while requesting input of the associated values. The variables to be shown on a two-variable map were stored as separate files.

The third step in assembling a choropleth mapping package consisted of writing a module through which a user could actually produce a map. The developed program included a menu driven main loop that called several subroutines to accomplish the following:

1. Read in a basemap file.
2. Read the data files.
3. Compute value range interval categories.
4. Set colors to the interval categories.
5. Draw the outline map.
6. Paint each area unit with the proper color.
7. Quit and close all files.

Steps 1 and 2 were simple disk filing operations which read the previously digitized basemap and the data into appropriate arrays. The identification of the value range

intervals in step 3 involved inputting the number of categories (2 to 4 for each variable) and then selecting a "manual" or "automatic" option. "Manual" allowed the user to identify the value range break points while "automatic" caused a programmed routine to develop the categories. The "automatic" algorithm compensated for skewed data and attempted to include approximately the same number of observations within each interval while maintaining a standard range width for all but two of the intervals. Top and bottom interval ranges (ie. from the minimum value to the lowest break point and from the maximum value to the highest break point) were allowed to be of variable width and different from that of the standard width of the other interval ranges.

Color selection (step 4) was from a pre-established gradient of light to dark shades in a red hue for the first variable and blue for the second. Once mapped, the hue bases could be exchanged or combined with green. Choropleth mapping was performed by a routine which painted each area unit polygon of the displayed basemap with a color value matched to the proper range interval for the combined variables.

RANGE INTERVALS AND COLOR SETTINGS

Computerized two-variable choropleth mapping is little different from single variable mapping except for two tasks: setting the class intervals and determining the color hues. Range intervals based on a bivariate distribution can be developed in several ways. Eyton described the methods and produced two maps combining income and education levels for counties in the United States (Eyton, 1984). The first map included classes based on a simple rectangular 3X3 red and blue frequency count matrix, while the second reduced the matrix dimensions to 2X2 and added a category corresponding to areas computed to be within a 50% equiprobability ellipse about the income-education relationship. Olson described criteria for effective interval development and legend representation, while Tobler and Eyton showed that classification is not necessary (Olson, 1972; Tobler, 1973; Eyton, 1984).

Intervals based on simple frequency counts are used for this study. The class generating algorithm attempts to minimize the variation in the number of observation grouped into each category while maintaining a uniform interval, but it does not compute probability estimates.

Minimizing the inter-class count variance for the separate variables on a two-variable map does not ensure representation in all categories. Perfectly correlated variables, for example, might have equal count representation among the classes for each of the separate variables, yet most of the cells of the rectangular matrix would include no observation at all. Of the 9 cells in a 3X3 legend, only the colors of the 3 diagonal cells would appear on the map.

The Amiga raster display maintains color information in a

maximum of 5 bitplanes, and binary coding of the bitplanes allows complete control of up to 32 different colors on the screen at one time. The colors are produced by separately setting hexadecimal red, green, and blue software switches, so the displayed colors can be drawn from a potential 4096 different shades (16X16X16). Several programming languages implemented on the Amiga provide access to the color controls and bitplanes through commands to library functions imbedded in the executive and operation systems. BASIC, for example, includes a Palette command for selecting the colors directly. Added display coloring on the Amiga can be achieved with sprites, a half-bright mode and a unique hold and modify control. These options allow concurrent display of the full 4096 colors, but their access and control is more difficult.

Several color settings were examined in reference to accepted criteria for good legend display (Olson,1972). Manipulation of two color registers (red and blue) proportionally to the two variables while holding the other register (green) constant created colors which provided appropriate gradient change, but the diagonal neutral tones did not range from black to white. If the green setting was at a minimum value, the tones along the diagonal of positive correlation progressed from black (lowest class for both variables) to deep lavender (highest for both variables). The more pleasing black to white diagonal transition was achieved by increasing the setting for the non-active color (green) proportionally to the oblique distance from the corner of lowest combined values (black). The black to white diagonal shift, usually requiring complementary colors, was thus obtained with primary colors.

CONCLUSION

As demonstrated, two-variable mapping is easily performed on a personal computer and at practically no cost once the initial investments have been made. Though the Amiga is not the only nor the best color graphics system, it is very inexpensive when compared to computers with comparable capabilities. What was done on the Amiga can also be accomplished on the Atari ST, Mac II, or the new PS II line from IBM. Two-variable mapping on a computer allows a cartographer to overcome the drawbacks of high cost and fragmented thematic demand which are associated with conventional production processes. The new technology available to geographers makes accessible previously impossible options for analyzing spatial data.

REFERENCES

- Anderson, J. (1985) "Amiga," Creative Computing, 11 (9): 32-41.
- Eyton, J. R. (1984) Complementary Color, Two-Variable Maps," Annals of the Association of American Geographers, 74: 477-90.
- Meyer, M., Broom, F., and Schweitzer, R. (1975) "Color

Statistical Mapping by the US Bureau of the Census," The American Cartographer, 2: 100-17

Olson, J.M. (1972) "Class Interval Systems on Maps of Observed Correlation Distributions," The Canadian Cartographer, 9: 122-31

Olson, J. M. (1981) "Spectrally Encoded Two-Variable Maps," Annals of the Association of American Geographers, 71: 259-76.

Tobler, W. R. (1973) "Choropleth Maps Without Class Intervals ?" Geographical Analysis, 3:262-65

Williams, G., Edwards, J., and Robinson, P. (1985) "The Amiga," Byte, 10 (8): 83-100

UPDATING URBAN STREET NETWORK FILES WITH HIGH RESOLUTION SATELLITE IMAGERY

L. Li, G. Deecker, K. Yurach
Geocartographics Division, Statistics Canada
Ottawa, Ontario, K1A 0T6

J. Seguin
Habitat and Land Use Division,
Canadian Wildlife Service, Environment Canada
Ottawa, Ontario, K1A 0H3

ABSTRACT

Digital street network files, such as Statistics Canada's Area Master File (AMF), are of increasing importance in mapping and GIS applications for the Census, municipal operations and in vehicle navigation. Rapid urban growth makes it a challenge to keep the files up-to-date in a cost effective manner. The heightened interest in digital network files has also raised questions about the geometric and positional accuracy of existing files.

Statistics Canada in cooperation with Environment Canada is conducting research to assess the feasibility of using high resolution satellite imagery with visual interpretation techniques to update urban street network files and identify newly urbanized areas. Excellent results have been achieved in the delineation of arterial roads and subdivisional collectors. Approximately 86% of subdivisional streets were successfully delineated in newer residential areas within the study sites. Older subdivisions with well treed streets and industrial-commercial developments were problematic. Positional accuracy of 16 metres was attained with off-nadir SPOT-PLA imagery.

INTRODUCTION

The recent availability of high resolution imagery from the SPOT satellite have led to increased interest in the use of satellite data for large scale mapping applications, (Thirwall et al, 1988; Begin et al, 1988), and detailed land use assessments, (Buchan and Hubbard, 1986; Milazzo and DeAngelis, 1984; Hernadez et al, 1984). Mapping of land use change and rural to urban land conversions are topics of interest to both Environment Canada and Statistics Canada. Environment Canada has been monitoring land use changes in the rural urban fringe of major urban centred regions across Canada for the last twenty years. Statistics Canada requires an up-to-date geographic base to support the Census and other surveys. These complementary interests led to cooperative evaluation of the potential of high resolution satellite imagery, SPOT and TM, to support these applications.

This paper reports on the technical issues, methodology and results from the cooperative study. Specifically, it evaluates the utility of SPOT and TM transparencies, with visual based interpretation, for updating digital street network files, such as Statistics Canada's Area Master File (AMF). The following issues are addressed:

1) comparative merits of Landsat-TM, SPOT-PLA and SPOT-MLA for the

task,

- 2) feature resolution - what features can be identified and delineated. How often are they correctly identified, and what are the magnitude for errors of omission and errors of commission,
- 3) factors which complicate interpretation,
- 4) geometric and positional accuracy of results, and
- 5) compatibility with existing AMF.

Study Sites

The problem of keeping street network files up to date is most acute in the urban fringe of major urban conurbations where development pressures are most concentrated. Two study sites covering a variety of land uses and development characteristics in the urban fringe of Ottawa and Toronto were chosen for studying the limitations of image data and the factors which affect interpretation accuracy.

The Town of Orleans, one of the fastest growing satellite communities in the national capital region surrounding Ottawa, is typical of many expanding suburban communities across Canada in terms of dwelling characteristics and land cover mix. The community is characterized by a small core of older residential neighbourhoods with houses of 25 years or more in age, aligned in a modified grid street pattern, in the mid-eastern part of the study site. A commercial strip exists along the two main thoroughfares. Surrounding the established core are more recent residential subdivisions, composed mainly of detached single family houses with some row houses. These were built within the last twenty years, and are typified by curvilinear streets. Large areas of new development are located in the southern portion of the town.

The second study site is a portion of the suburb of Rexdale, in the northwest section of Metropolitan Toronto. This area encompasses a diversity of land uses including warehousing, light industries, and residential neighbourhoods of varied ages, dwelling types and tree cover. The older residential neighbourhoods, in the middle eastern part of the study site, is predominantly detached bungalows, approximately 30 to 50 years old. Mature trees line the streets with the houses uniformly set back from the curbs. Progressing northwards, one encounters a mix of row houses, semi-detached and detached houses that are 10-20 years old. Mature trees are generally absent from these areas, except in some of the row house developments. At the northern edge of the site are large, irregularly shaped single family houses, 5-10 years in age. This site provides a complex land use and buildings mix to explore the limitations of the data set.

METHODOLOGY

Inputs

The following data were used in the study:

- 1) SPOT-PLA (10 metres spatial resolution) and MIA (20 metres spatial resolution) test scenes were acquired for the Town of Orleans. The

PLA and MLA scenes, dated August 5, 1987, were taken with look angles of 27.89 and 28.27 degrees, respectively. For Rexdale, a regular system geocoded PLA image, dated June 4, 1987 with a look angle of 2.60 degrees, was used to cover the site.

- 2) Landsat TM (30 metres spatial resolution) colour composites, Bands 2, 3 and 4, dated June 186 for Orleans and May 186 for Rexdale.
- 3) 1:10K topographic maps for Orleans and Rexdale to serve as reference bases for creation of 1:20K base maps for image interpretation and delineation of roads. The maps for Orleans and Rexdale are derived from 1979 and 1982 air photos, respectively.
- 4) 1:2K engineering maps to serve as benchmarks for positional verification. The engineering maps for Orleans were compiled from May 1983 photos, while the Rexdale coverage was compiled from 1982 photos.
- 5) 1:5K, 1988 orthophotos of the Rexdale study site to provide ground truth data.
- 6) Area Master File (AMF) for the City of Gloucester which covers a large part of Orleans study site. This AMF was last updated in 1987 using municipal maps.

Procedures

- 1) Selected main arterial roads from 1:10K topographic maps were digitized using ARC/INFO and replotted at 1:20K to serve as a reference base to guide image interpretation.
- 2) Image interpretation:
 - a) Roads were identified and delineated in pencil onto individual 1:20K reference maps from the TM, SPOT-MLA and SPOT-PLA transparencies, in their respective sequence, for the Orleans study area. Visual interpretation was carried out using the PROCOM-2 Image Analysis System, an analogue projection device. Visible landmarks, such as schools, hospitals, parks, prominent buildings, etc. were also noted on the base map.
 - b) The same procedure as (a), but using only PLA imagery was then carried out on the Rexdale study site.
- 3) Feature resolution assessment - the delineation results were digitized and plotted at 1:2K and 1:5K, and compared with up-to-date air photos, topographic maps, engineering maps and the AMF to determine the accuracy of the captured features and the causes of error. Field checks were conducted where uncertainty existed.
- 4) For Orleans, the delineation results were transformed from UTM to three degree Transverse Mercator projection, then plotted at 1:2K for overlay and comparison with the engineering maps to assess the positional accuracy of the results. Positional error was evaluated by measuring the offset between the position of road intersections on the delineation overlay and the position of the same intersection on the engineering map.

5) AMF-SPOT compatibility was analyzed by overlaying the AMF with the SPOT delineation results. The assessment focused on the geometry, position, completeness and presence of errors of omission and commission.

RESULTS

Imagery Comparison - PLA vs MLA vs TM

A comparison of SPOT-PLA, MLA and LANDSAT-TM images was conducted to determine the most suitable imagery for the task of updating urban street network files.

Visual examination of the TM scenes indicated that delineation of residential roads within subdivisions would not be possible for many urban neighbourhoods. The insufficient spatial resolution resulted in a mottled pattern where the roads were indistinguishable.

On both SPOT MLA and PLA scenes of the Orleans study site, roads and some landmarks were clearly visible on cursory examination. However, when actual delineation of the features was attempted, some difficulties were encountered in determining the existence of some street segments.

Figure 1 - Overlay of the Orleans SPOT-PLA and MLA Delineation Results

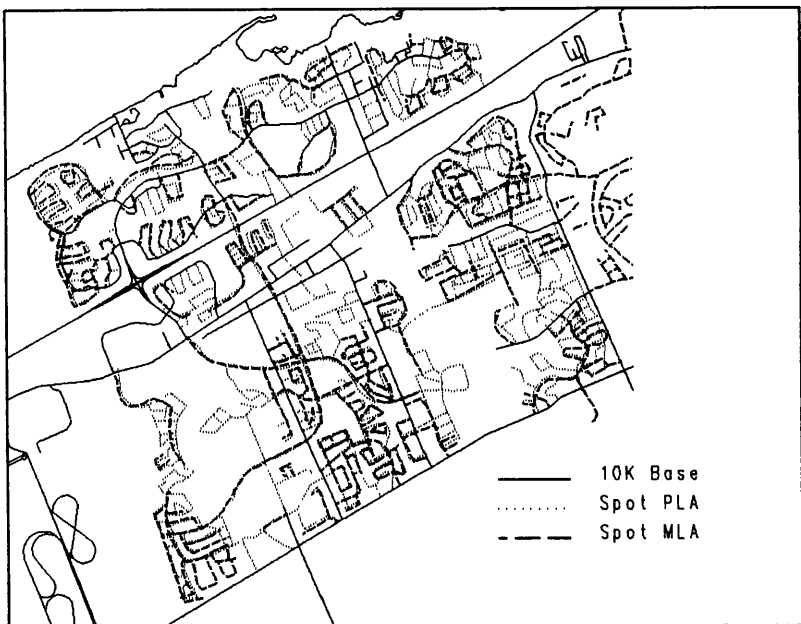


FIGURE 1 illustrates the achieved results. The superior resolution of the PLA imagery is evident in the high degree of completeness and better geometry of the captured road network. Quantitatively, the PLA imagery enabled correct delineation of approximately 86 % of the total of 93 kilometres of the roads within the Orleans study site. 54 road

segments were missed totaling 11.7 kilometres, and approximately 28 segments, totaling 2.9 kilometres were incorrectly added. For the MLA imagery, approximately 64 % of the roads were correctly delineated. 101 segments, totalling 32 km were missed, and approximately 13 segments, totalling 1.2 km were incorrectly added.

Large buildings and building sites, such as warehouses, offices and light industries, are visible where there is adequate contrast with the adjacent area. However, positive identification of the purpose of the building - hospital, warehouse, etc. is seldom possible. Oval sport tracks which are often associated with schools and community recreation centres, are identifiable in areas of sufficient contrast. Quantitative assessment of the comparative resolution of landmark features was not carried out.

Resolution of Roads with SPOT-PLA

The features of prime importance are roads for street network applications. The key issues are: 1) whether the features can be correctly delineated and classified and 2) how often features are incorrectly identified and delineated, either through improper classification, omission of features, or addition of features where none really exist.

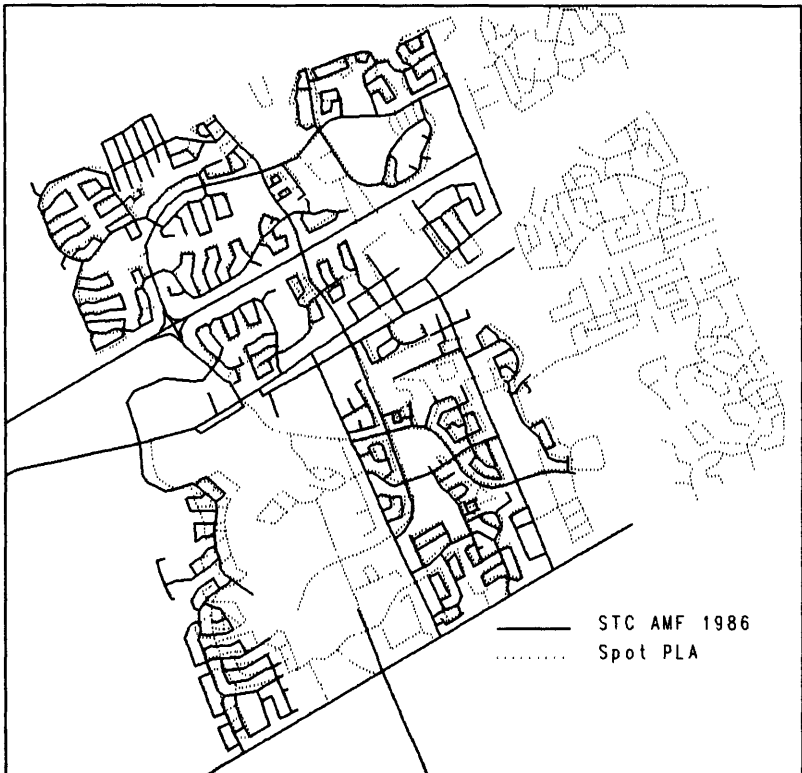
As evident in **Figures 2 and 3**, widely divergent results were obtained from the two study sites for the delineation of roads. The differences in the interpretation accuracy can be attributed to a number of factors including: the age of the neighbourhoods; dwelling types; amount and the maturity of the tree cover; alignment and shape of the houses; and presence of walkways in residential neighbourhoods. In industrial-commercial complexes, the landscaping and land cover significantly influence interpretation accuracy.

Comparison of the delineation results with the 1986 AMF for Orleans, **Figure 2**, indicates near complete capture of the road network. Approximately 86% of roads were correctly delineated with near perfect capture of all arterials and subdivisional collectors. Errors of omission and commission were evident in the mapping of subdivisional streets, with errors of omission being more prevalent due to the conservative interpretation approach used. Common errors included improper closure of crescents and incorrect joining of one crescent with another.

A higher incidence of error was found in the middle eastern portion of the study site where a mixture of older residential dwellings, detached houses from 5-20 years old and several apartment buildings are present. The errors include mistaking apartment parking lots and the long linear roofs of the apartment buildings as connecting road segments. The tree lined streets of the older residential area led to some misinterpretation of the ends of streets and incorrect connection of other streets. Use of imagery from early spring or late fall, rather than the June scene of the current study, would help to reduce this problem.

Figure 3, an overlay of the SPOT delineation results with the topographic base map, provides an indication of the lack of congruence between the delineated roads and the actual roads in for the Rexdale study site.

Figure 2 - Overlay of SPOT-PLA Delineation Results on the Orleans AMF



In the southeastern quadrant of the site, an older residential with area of 30-50 year old bungalows, the mature deciduous trees obscuring parts of the streets and linearly aligned houses of similar shapes and often with paved driveways in the space separating adjacent buildings led to mistaken identification of the rows of roofs as roads. The tree canopies also interrupted the expected pattern which made interpretation more difficult.

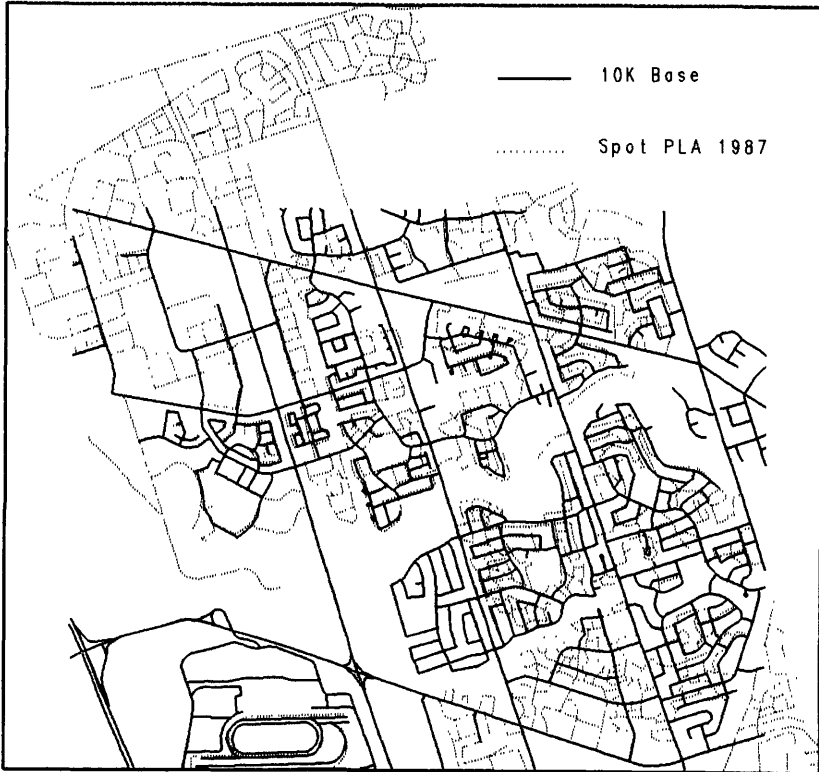
The central part of the study area is characterized by a mixture of semi-detached houses, detached houses and several row house developments, approximately 15-25 years old. In the row house complexes, success was extremely poor as the long roofs of the row houses, coupled with the network of walkways and mature evergreens resulted in the delineation of a confused pattern of streets where none existed.

The industrial area, in the southeast and northwest of the site, is characterized by expanses of pavement surrounding buildings. The difficulty in differentiating these pavement areas and the large roof of the commercial-industrial buildings from paved road surfaces resulted in numerous misplacement of roads and omission of existing streets.

At the northeastern fringe of the site are two 5-10 year old residential subdivisions. These subdivisions are similar to those

found in Orleans with predominantly large detached single family houses, although their street patterns are somewhat unusual. The results achieved was similar to that of Orleans, where approximately 86% of the roads were correctly delineated. Again prevalent errors were omission of short street segments, mostly short dead-end streets, and incorrect connection of crescents, due mainly to misinterpretation of paved walkways or aligned light coloured roofs as streets.

Figure 3 - Overlay of SPOT-PLA Delineation Results on the 10K Rexdale Topographic Base



Resolution of Road-Related Features

Associated with roads are point and line features such as overpasses, underpasses, bridges, tunnels and ramps which are important for many street network applications, e.g., ambulance dispatch. A number of overpasses/underpasses, bridges and ramps are present in Orleans and Rexdale. Most ramps to major expressways are clearly visible by their cloverleaf pattern, however, where there are no grassed medians between the ramp and the roadways identification is more difficult. Overpasses/underpasses can be identified by determining which roadway is obscured at the intersection. However, only the places where major expressways pass over arterial roads and vice versa were identifiable. Likewise, only major bridge crossings of significant watercourses, such as the Humber River in the study area, can be interpreted. Minor

watercourses which may be bridged or spanned with a culvert are not discernible.

Resolution of Landmarks

Landmarks are useful in street network files to assist users in spatial orientation. A variety of point, line and area features are captured in the AMF to provide census enumerators with landmarks to orient themselves. These include prominent buildings, like hospitals, school, churches, golf courses, cemeteries, hydro lines and railways. From the Rexdale image, some large buildings or building sites, three of five sports tracks, and horse racing ovals can be identified. However, correct classification of buildings as schools, colleges, hospitals or other institutions is not possible from the imagery, although assumption of relationships, such as the association of sport tracks with schools may make preliminary classification possible. High tension hydro transmission corridors and multi-track rail corridors are distinguishable.

Positional Accuracy

The positional accuracy of high resolution imagery for mapping has been a topic of much interest, especially within the topographic mapping community. Recent research indicates that X Y accuracies in the order of 5-15 metres can be expected depending on the scale of mapping (Thirwall et al, 1988; Begin et al, 1988(2)). The studies have generally relied on topographic maps at the same or slightly larger scale as the benchmark for comparison. As well, the focus have been on nadir looking scenes.

In this study, both a near nadir looking scene for Rexdale and a scene with an extreme look angle, Orleans, were assessed. Engineering maps with positional accuracies of better than one metre, were used as the measurement benchmarks.

From the Rexdale site, some promising measurements were recorded, however, the above-noted inability to accurately delineate roads within older residential subdivisions and industrial-commercial areas precluded fair assessment of positional accuracy.

The Orleans results, from a sample of 96 points, yielded a mean positional error of 16 metres, with a maximum error of 46 metres and a standard deviation of 0.443. Breakdown of the spatial distribution of error by neighbourhood characteristics showed no significant variation.

CONCLUSIONS

The results to-date indicate that for updating urban street network files, SPOT-PLA images can be useful, with approximately 86% accuracy in road identification in newer residential areas. SPOT-PLA appears to be most suitable for the task, as opposed to SPOT-MLA or LANDSAT-IM, although the combination of SPOT-PLA and MLA may be helpful to improve discrimination of vegetated areas from road surfaces in some areas.

With SPOT-PLA, primary highways, arterial roads and subdivisional

collector streets can be confidently mapped. For subdivisional streets, excellent results can be attained for new single family detached housing developments. Correct results are less certain for row house developments without interior streets and in areas adjacent to large buildings and large paved surfaces. Older neighbourhoods with a mature tree cover and narrowly spaced, linearly aligned houses are problematic, especially when summer scenes are used for analysis. Areas having extensive paved surfaces and roof areas, such as large industrial-commercial complexes are confusing, since there is little difference between these hard surfaces and paved streets.

Identification of road related features, such as overpasses and bridges, and landmarks, such as schools, hospitals and golf courses, etc. are possible on occasion. Development of contextual rules to assist interpreters would be of benefit, not only for visual interpreters but also for automated digital processing.

In terms of positional accuracy, the results indicate that with careful referencing, off-nadir imagery can yield acceptable results with only minor degradation of accuracy.

Overall the results are encouraging and confirms the utility of high resolution satellite imagery for updating urban street network files and monitoring changes in the urban-rural fringe. Research is continuing to address some of the issues which still remain to be resolved before this application can be brought to fruition. Notably, the adequacy and reliability of image coverage must be assessed. Appropriate dates/seasons of the image to optimize feature identification would benefit from further study. Applicability of the techniques in higher relief urban centres and the effect of street pattern complexity must be further evaluated. Finally, once feasibility is better established, a detailed cost benefit analysis must be carried out to determine the end viability of this application.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the cooperation of Mr. B. Serson, Regional Municipality of Ottawa-Carleton and Mr. R. Smith, Toronto Central Mapping Agency for providing base maps and information for the study areas, and Mr. J. P. Parker, Geography Division, Statistics Canada, for availing the Gloucester AMF for use in the study. The advice and editorial assistance of Mr. C. Duguay of the Geocartographic Division, Statistics Canada is also much appreciated.

REFERENCES

- 1) Bégin D., Y. Boucher, J. Brodeur, J. Gauthier, C. Girard, R. Hastie, J-P Lemieux, L. Ouellette and M. St-Pierre, (1988), Contenu cartographique des images SPOT, paper presented at Symposium International sur les applications topographiques des données SPOT (Oct. 1988), Sherbrooke, Quebec.
- 2) Begin D., Y. Boucher, J. Brodeur, C. Girard, D. Lapierre, J-P Lemieux, J. Gauthier, R. Hastie, V. Kratky and M. Wijk (1988), Précision géométrique des données SPOT, paper presented at Symposium International sur les applications topographiques des données SPOT (Oct. 1988), Sherbrooke, Quebec.

- 3) Buchan C.M. and N.K. Hubbard, (1986), Remote sensing in land-use planning: an application in west central Scotland using SPOT-simulation data, Int. J. of Rem. Sensing, vol. 7, no. 6, pg. 767-777.
- 4) Hernandez M., P. T. Nguyen and A. Ballut, (1984), Change detection in urban areas: a SPOT simulation experiment, Technical Papers 1984, World Conference on Remote Sensing, Bayreuth, West Germany, Oct. 1984.
- 5) Milazzo V.A. and R. DeAngelis, (1984), Applications of simulated SPOT data to mapping land cover patterns and changes in an urban fringe environment, SPOT Applications Handbook, Proceedings of the 1984 SPOT Symposium, SPOT Image Corp., Washington, D.C.
- 6) Thirwall S.L., C. Galipeau, S.D. Melvin and H.D. Moore, (1988), Comprehensive evaluation of high resolution satellite imagery for map revision and change detection, prepared for Surveys and Mapping Branch, Energy, Mines and Resources and Supply and Services Canada, Contract No. 23232-6-1326/SQ, Ottawa, Canada by Gregory Geoscience Ltd., Ottawa, Ontario.

EDUCATION AND TRAINING IN GIS: THE VIEW FROM ESRI

Tony Burns and Jim Henderson

(I) GIS Training: a Voyage of Discovery

GIS education is very much a voyage of discovery, a voyage of inquiry into the fields of geography, cartography and computer technology.

To properly learn about GIS we must first learn about geography and what it implies. Geography is a science that deals with the earth and its life, especially the description of land, sea, air and the spatial distribution of geographic phenomena; plants, animals, man, his cities with reference to the mutual relations of these diverse elements.

Early in the Second World War, President Roosevelt asked his audience, during one of his "fireside chats" by radio, to look at a world map while he explained the threats to the United States of hostile encirclement if the Axis powers were to win. Similarly, President Kennedy told us of the strategic importance of an unknown area called Laos and Vietnam. In fact, both Presidents were discussing geographic concepts of location and interrelationships between parts of the earth.

More recently, as inhabitants of various cities throughout the United States, we find ourselves concerned with such matters as floods and water shortages, toxic and hazardous wastes, water pollution, earthquake risks, chaotic growth and congestion in our metropolitan area, depletion of wildlife habitats and wetlands, and legislative reapportionment. But all too seldom do we clearly understand these problems in relationship to the surrounding geography. We clearly need to understand how people and places interact.

For hundreds of years we have used maps as a visual geographic tool. Maps are the language of geography. In producing maps the cartographer sets out to accurately map geographic phenomena. In one small area of one vast country there are scores of features that are full of wonder and mystery. I want to know with precision the location of these things I see before me relative to the geography around and related to them. I want to know their shape, size, composition, and interrelationship. I want to unravel the interrelationships and understand the mystery. As part of my inquiry I want to answer three essential questions:

- Why are things located in particular places;
- How do these particular places influence our lives; and,
- How do these things interrelate and how can I better understand these interrelationships.

Today's technology has let us explore new ways to understand geography and answer our questions. We can look at a map in new ways. If we can read its language we can begin to understand our world, our cities, and our impact on them. We are shapers of the landscape both for good and bad. There are many ways of looking at places, many perspectives. I can define a place as a dot on a map; a unique mathematical coordinate on the sphere of the earth, or I can step back further and see the pattern created by many dots. At this level I see places not as isolated features but relative to one another. I see them in context; cities connected by vast networks of roads, some dots larger than others, nations of people and ranges of mountains.

I can also step closer and see that the dots are much more complex. Now I begin to see the shapes that make up cities. I see the specific mathematical features that make up my map; dots or points now represent such features as utility poles, lines are now street centerlines, property lines and utility lines, and shapes or polygons are now building footprints, lakes and easements. Computers are tools that let us effectively and efficiently deal with geography and the language of part of previous maps.

As teachers we must clearly convey the concepts of understanding a spherical world as a set of points, lines and polygons on a map. The concepts of scale, map projection, and manipulation and analysis of map features via a set of GIS software tools are essential to understanding spatial relationships. Only then can the resulting maps be of use in decision making and problem solving.

The challenge facing today's GIS teachers is to carefully balance the concepts of geography, dealing with maps as geometric features upon which mathematical functions are applied, and dealing with a technology of software tools operating on a set of electronic circuitry. Our goal is to uncover patterns and themes, making inquiries of a complex nature.

Let's look at it in another context. A Geographic Information System is:

- Geographic:** Spatial Data about geography; maps: the language of geographers.
- Information:** The analysis and synthesis of data to answer questions, solve problems, and make decisions.
- System:** The integration of hardware, software, people, data, and administration to operate as a whole.

At ESRI we believe we have carefully developed our training program to incorporate these elements. Let's explore the process and examine the philosophy behind our approach.

(II) Education and Training: Background and Issues

Geographic Information Systems, in one form or another, have been around since the late 1960's, but only in the past few years have they been introduced into the mainstream of computer applications. In those few years the growth of GIS sales has been tremendous. The GIS now finds a place in many areas of business, government, and society where people not only are unfamiliar with its technology, but may also be approaching computers for the first time.

This situation poses a special challenge for the GIS vendor in training the new user. By its inherent nature, GIS technology is founded upon a set of basic geographic and spatial concepts that incorporate a holistic view of the world. These concepts are typically not part of everyone's academic training or professional practice. In order to make effective use of a GIS, one must understand these basic concepts, for they are the foundation of its functionality. On top of this, the GIS operates within the context of fundamental Automated Data Processing principles which may be foreign to many prospective users.

Traditionally, universities have prepared many people for their professions, but GIS is still an emerging discipline in search of its identity. As a result, academia has not yet fully defined the field of study or the curriculum for its preparation. This places a further burden upon the GIS vendor to address basic educational issues in training. This situation is slowly changing as the academic GIS discipline evolves.

But in the meantime, the GIS vendor is faced with several challenging issues in its training. The first is defining the relationship between technical training, fundamental concepts, and basic education. The second issue is determining the vendor's role and responsibility in providing a minimum background knowledge as prerequisite to its technical training. Finally, how to address the need for advanced and continuing training for users as they mature is an important issue for the GIS vendor who is committed to supporting its users.

What follows is the perspective taken by ESRI with respect to these issues. ESRI's philosophy of providing strong support to its users is inherent in its comprehensive approach to training.

(III) Training vs Education

Within the general "learning environment" one can distinguish between two major divisions, "Training" and "Education." Each is based upon different principles and purpose which may represent opposite ends of the spectrum at times. Whereas education seeks to enable students to understand basic concepts, theories, and principles, training strives to make the trainee proficient in using the functions of a particular tool. Similarly, education enables students to apply concepts, principles, etc. in a wide variety of environments, while training aims to enable trainees to apply functions to specific situations. Training and education may be contrasted in other ways.

<u>TRAINING</u>	<u>EDUCATION</u>
<ul style="list-style-type: none"> • Specialized Instruction • Short Term Time Frame • Concentrated Attention • Intense Delivery • Practical Emphasis • Performance Skills • Behavioral Change 	<ul style="list-style-type: none"> • General Instruction • Long Term Time Frame • Dispersed Attention • Measured Delivery • Theoretical Emphasis • Knowledge Acquisition • Synthesis of Ideas

As was pointed out in the previous section on issues, the increasing trend toward a training audience with a naive understanding of basic geographic and spatial analysis concepts forces the GIS vendor to address GIS education directly as a prerequisite to training. The vendor must not assume this prerequisite education will be the responsibility of someone else nor be acquired prior to the vendor's training. Therefore, it is imperative for the vendor to incorporate at least a minimal level of basic education in the design and delivery of its technical training. Without it the vendor's training will be ineffective and the client unsuccessful. At ESRI, we recognize the importance of education in the design and delivery of all our technical training.

(IV) The GIS Training Course at ESRI

(A) What is it?

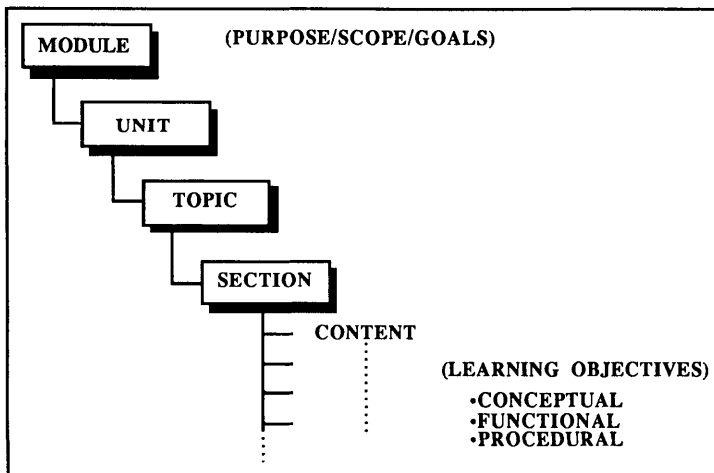
GIS training courses at ESRI are designed around four basic components. The first and most important component from the standpoint of education is that of "concept." It incorporates those fundamental ideas, theories, and principles which form the foundation for the second component, system functionality. ARC/INFO functions are the implementation of basic concepts, theories, and principles through well structured software engineering. A significant portion of technical training is concerned with

attaining a basic understanding of the system functions. This understanding is a prerequisite for the third component of the training, procedure. This part of the GIS training course concerns itself with acquiring the skills needed to effectively use the functions of the system. In order to achieve this the training must involve a great deal of hands-on exercise with the tools. It must not only demonstrate the correct use of each tool, but must also show the proper sequence of their use. A primary goal of this part of the training is the enable the trainee to make intelligent choices in the use of the system functions. Closely related to this is the last component, the application of the system. Here the goal is to prepare the trainee for the transition from the classroom to the real world. To achieve this objective the GIS training must relate the knowledge and skills acquired by the trainee to real world situations. In other words it must make the training relevant.

A well-designed and effectively delivered GIS training course will give the trainees the knowledge and skills they need to feel confident in using the system upon their return to their jobs. It will also prepare them to begin the process of learning more on their own. ESRI training courses are directed toward achieving these results.

(B) How is it Designed?

The model for the design of ESRI training courses is based upon well established instructional design principles. Professional training developers apply these principles in the design and development of all ESRI training. The instructional design model incorporates specific "learning objectives" derived from an analysis of the "audience" and their needs. The learning objectives are expressed in terms of specific skills to be acquired or levels of knowledge to be attained. In other words, the focus of the training is based upon what the trainee will be expected to do or know upon completing the training.



INSTRUCTIONAL DESIGN MODEL

(C) How is it Delivered?

Just as we have a model for the design of training, we have a model for its delivery. The two basic modes of training delivery are instructor-lead/classroom and self-paced/stand alone. Each mode has its place in the overall delivery strategy for the training program.

The classroom delivery model incorporates well established and accepted techniques and media. The intent of the model is to deliver each "section" of the training course in the particular sequence and method to be most effective in achieving the learning objectives.

<u>MEDIA</u>	<u>DELIVERY METHOD</u>
Video and/or Lecture	(0) Gaining Attention/Getting Started
"	(1) Presentation of Concepts
"	(2) Relevance and Relationship to Other Sections
"	(3) Presentation of Technical Procedures ("How to...")
Exercise or Tutorial	(4) Acquisition of Skills ("Hands-On") (a) Specific (Command Usage/Simple Procedures) (b) Comprehensive (Integration of Knowledge/Skills)
Lecture and Discussion	(5) Review/Discussion/Evaluation ("Transition" to Next Section)
Self Paced Materials	(6) Job Aids/Workbooks/Guides

DELIVERY MODEL

(V) The GIS Training Program at ESRI

The approach to planning and implementing a comprehensive training program involves a well thought out strategy for designing the curriculum. Just as there are basic differences between education and training with respect to instruction, so too are there differences in curricula.

<u>TRAINING</u>	<u>EDUCATION</u>
<ul style="list-style-type: none">• Specialized Courses• Narrow Focus• Customized Instruction• Technical Orientation• Application	<ul style="list-style-type: none">• General Courses• Broad Focus• General Instruction• Professional Orientation• Interdisciplinary

The training program at ESRI encompasses a curriculum of courses designed around a wide variety of client needs. The strongest emphasis is on building a solid foundation of basic system training. There are several courses devoted to the general understanding of the software modules. These basic level system training courses are normally provided with the sale of each software module.

Beyond the basic level of training are courses designed to meet the needs of clients for continuing education and more advanced system tools for specific applications or to achieve higher productivity. Typically these courses follow on several months after the basic courses and are required by a more selective audience. The basic level courses are normally required as prerequisites.

Another audience which must be considered is one which doesn't require an in-depth knowledge of the workings of the software modules. Instead, they need to know the basic concepts underlying the system, its capabilities, and some guidance in managing its applications. This audience typically includes managers, administrators, project directors, and support personnel. The ESRI training curriculum includes an "Introduction to GIS" course to address this need.

Finally, as clients mature they build a vast base of experience that can be invaluable to less experienced users. It is important to tap this tremendous resource of knowledge and skills in a way that makes it accessible and useful to others. The "Application Seminar or Forum" format is a very good way to structure this transfer of experience from one to another. At ESRI we feel it is our role to organize and implement these seminars/forums, but the instructors must be users who possess the experience in the particular application. In this way, as a vendor, we operate much like an "extension service."

In summary, the ESRI curriculum is structured on several levels, each designed to meet specific client training needs.

LEVEL 0	Introduction to GIS
LEVEL 1	Basic ARC/INFO System Training Basic TIN System Training Basic NETWORK System Training Basic COGO System Training
LEVEL 2	Database Design Applications Programming Cartographic Production Geographic Analysis Processing Techniques System Programming Systems Administration
LEVEL 3	Application Seminars/Forums

(VI) Conclusions

In conclusion, the tremendous growth in GIS sales has placed the GIS vendor in a position of having to provide more comprehensive training than in the past. As GIS technology reaches out to a larger audience, the level of understanding of fundamental GIS concepts and computer operations becomes less sophisticated. In order to assure success in its training, the GIS vendor must incorporate basic education in its training program. As its clients mature in their experience with GIS technology, the GIS vendor must recognize the need for more continuing education and advanced training. Ultimately, the GIS vendor has a role in all levels of GIS education, training, and technology transfer.

GEOGRAPHIC INFORMATION SYSTEM TEACHING AT ITC

Ms.J.Drummond, Dr.J-C.Muller, Dr.P.Stefanovic.

Department of Cartography, International Institute for
Aerospace Survey and Earth Sciences (ITC), POBox 6, 7500AA
ENSCHDEDE, The Netherlands.

ABSTRACT

ITC was founded in 1951 as a photogrammetric training centre, but now has five departments: Aerospace Data Acquisition and Photogrammetry (ADAP); Cartography; Land Resource Surveys and Rural Development; Earth Resource Surveys; and Urban Surveys and Human Settlement Analysis. ITC students are mostly from developing countries, and already have professional expertise; they arrive with specific needs which must be met by our education program.

By the mid 1980's GIS teaching was established in all departments, and for example, graduate Cartography students will learn map design and production only within the GIS environment and graduate Photogrammetry students have as a central theme the capture and processing of photogrammetric data within the GIS context. Beyond this evolution, a specific LIS course having three streams: multipurpose cadastre; urban applications; and rural applications, began in 1985. LIS Course participants usually come from cadastral, legal, rural planning, or urban planning professions and the course aims to "prepare participants to manage the design, implementation, and maintenance of geographic or cadastre based Land Information Systems".

ITC LIS Course participants often lack computer skills when they arrive in Enschede - so these are taught via GIS problems. Because course participants often return to organisations with low funding levels, much coursework uses microcomputers. Finally contact is maintained on returning home through the ITC JOURNAL and joint projects between ITC Staff and former course participants.

1.0 A WORKING DEFINITION OF GEOGRAPHIC INFORMATION SYSTEMS

In a recent paper by an ITC staffmember, which approached the problem of establishing a GIS from the viewpoint of information utilization [De Man,1988] it was stressed that GIS systems are similar to information systems in general in that they accept, process, present, update, modify, and combine data from a variety of sources - but that in the case of GIS the data has a locational attribute. It was also stressed that there are general feelings of scepticism towards all information systems in that they do not provide useful results automatically; this scepticism is not ignored at ITC, but is alleviated by the fact that good presentation of locational data has been achieved for many years through the techniques of Computer Assisted Cartography (CAC), and these techniques are easily incorporated into GIS. Both conceptually and practically a

GIS may be the superstructure on a CAC system, and this bottom-up approach finds some favour at ITC because of its practical usefulness. Negative characteristics of the bottom-up approach include the conflicting locational, attribute, and quality objectives of data gatherers resulting in data sets which cannot be combined; the conflicting time requirements of users who need 'their' data immediately and cannot wait for them to be prepared for general access in an information system; and institutional barriers which prevent the flow of data between information system users.

The top-down approach, which may be the ideal approach, also has its advocates at ITC [Jerie, Kure, Larsen, 1980]. The top-down approach begins with a decision by the political masters of the information users, the data gatherers, or both, to integrate their efforts. From this political decision will eventually flow consistencies in software, hardware, and data standards. Unfortunately the political will may not be there, or may not be there for long enough to ensure the establishment of a good GIS.

In 1984 when ITC's GIS Working Group established the syllabus for its LIS/GIS Course it was realised that the political environment within which a GIS is to be established has considerable bearing on whether or not the top-down or bottom-up approach is adopted. As our course participants, or students, come from countries representing the complete political spectrum (and no one political system dominates), then participants have to be aware of both approaches. ITC course participants are usually mid-career professionals, generally in their thirties or forties so they are keenly aware of the importance of compromise. The success of GIS in Burnaby, B.C., Canada, which began as a bottom-up development, achieved eye-catching success, and subsequently received top-down enhancement is a model with which many of our course participants identify.

In terms of organisational structure there is no LIS/GIS department in ITC. Taking the view that LIS/GIS is a coordinated collection of tools and technologies for geoinformation production, a GIS working group involving a wide variety of earth science disciplines was created instead. The common working basis of the working group finds its terms of reference within the following conventional definition:

"A Geo Information System is a system for capturing, storing, checking, integrating, manipulating, analysing, and displaying data which are spatially referenced to the Earth. This is normally considered to involve a spatially referenced computer database and appropriate applications software. A GIS contains the following major components: a data input subsystem, a data storage and retrieval subsystem, a data manipulation and analysis subsystem and a data reporting subsystem."

This definition is a foundation of the ITC LIS Diploma courses.

It should be noted that although the term LIS often refers more specifically to cadastre related information, the terms LIS and GIS are used quite interchangeably at ITC.

2.0 THE NEED FOR GIS EDUCATION

In the industrialised countries if an organisation does not possess staff with the right skills it must either:

- (i) recruit new staff from schools;
- (ii) poach staff employed in other organisations; or,
- (iii) have existing staff retrained.

But, in industrialised countries there is a general shortage of personnel with the skills to develop and manage GIS, thus schools have to establish courses to provide or retrain staff. It is because schools realise "large sums of money are spent by Government, commerce and industry, the utilities, the armed forces, and others in collecting and using it" (i.e. geographic information), "...much human activity depends on the effective handling of such information.." and, "it is the ability of...Geographic Information Systems...to integrate these functions and to deal with the locational character of geographic information" [CHORLEY,1987] that they are beginning to establish GIS courses. At the moment these courses are directed towards the development and management of GIS; they are essentially postgraduate courses.

In the industrialised countries technician level training is essentially on-the-job, however GIS skills are so removed from other skills acquired in secondary education that technician level training may be important as a foundation for GIS operators to further develop their skills on-the-job.

For GIS in non-industrialised or less developed countries (LDC's) staff hardly exist yet to be hired directly from schools or poached, so only the third option exists, namely staff retraining. Furthermore the training has almost always to be away from the home country.

In LDC's there may not yet be the same mass of geographic information in computer compatible form which is available in industrialised countries, but due mostly to rapidly expanding populations, there is the need to strengthen the managerial functions of government. These functions include planning, decision making, inventorying, and monitoring; all these functions are functions of GIS.

As well as the need to strengthen the managerial functions of government most LDC's lack assets, and may look to international bodies such as the World Bank to develop these. Although most LDC's lack assets, two assets which they have are the land and the people on it, and these are ideally suited to management by a GIS - and especially a Cadastral GIS. There is now a trend in organisations such as the World Bank to assist in asset development if an LDC demonstrates an intention or capability to manage its existing assets.

Under the type of pressure outlined in the preceding paragraphs, LDC's are now considering GIS training. This must usually take the form of staff secondment for one year to a training course in an industrialised country. Such staff are almost invariably graduate, but their most distinguishing characteristics are that they are already experienced professionals, usually in mid-managerial ranks, in secure employment, and often have a very good grasp of what their own country needs. It is these professionals who

form the core of ITC's LIS/GIS student body.

3.0 ITC's LIS/GIS COURSES

There are three groups of LIS/GIS courses at ITC:

1. Degree Courses (Ph.D. or M.Sc.);
2. Interdepartmental LIS Diploma courses; and,
3. GIS modules within other Diploma courses.

3.1 Degree Courses

Dutch Universities are closely monitored by the Dutch government, and so for administrative reasons the doctoral courses have to be given in conjunction with a traditional Dutch university and not by a Dutch International Institute (such as ITC) on its own. In practice this means a student will have a supervisor (or promoter - to use the Dutch terminology) from a university such as Utrecht, Wageningen, or Delft as well as a supervisor in ITC. Several ITC Ph.D. candidates are now preparing theses in the GIS domain.

The M.Sc.'s need not, legally, be given in conjunction with a Dutch university, however ITC has established a cadastre-based M.Sc. course jointly with the Geodesy Department of Delft University, for students who typically already have an ITC LIS Diploma with the cadastral specialisation. Students wishing to specialise in urban or rural applications of GIS, and who typically have already completed the LIS Diploma with urban or rural specialisation, may carry out their M.Sc. work within the ITC departments of Urban Surveys and Human Settlement Analysis or Land Resource Surveys and Rural Development. Students not having an ITC LIS diploma, but instead typically one in Cartography, Photogrammetry, Urban Surveys, etc., may also choose as a thesis title a topic clearly within the LIS/GIS sphere (and at the moment most do!), but on completion of their M.Sc. their knowledge of LIS/GIS is likely to be shallower than that of a student who has completed the ITC LIS diploma course. At ITC an M.Sc. course lasts about twelve months, and usually immediately follows a twelve month diploma course. The diploma course is not a prerequisite for the M.Sc. course, but very few students are accepted for direct entry to the M.Sc. The M.Sc. course consists of 500 hours of coursework, followed by thesis work taking about eight months.

3.2 LIS Diploma Course

The ITC Interdepartmental LIS Diploma course began in 1985. Its syllabus was the product of interdepartmental deliberation. It consists of three streams:

cadastral;
rural; and,
urban.

The aim of the cadastral stream course is to enable participants to establish a cadastral GIS at national or municipal level, for legal, fiscal and other purposes; to upgrade cadastral GIS; to expand an existing cadastral system into a multi-purpose cadastre or large-scale cadastral GIS to be used for title registration, valuation, and assessment, administration and social services, and the development of utilities, services, and transportation.

The aim of the rural stream course is to make participants familiar with available hard and software for spatial analysis and survey and the potential uses of these systems for resource management, development, and conservation; and, to enable students to evaluate techniques of data collection, processing, analysis, and presentation.

The aim of the urban stream course is to familiarize participants with the application of GIS as a vital tool to a city's strategy to improve the quality and control of urban planning and management; and, to familiarize participants with physical urban planning, traffic planning, education planning, and land management. [LINDEN, 1988]

The course can accommodate 35 participants, and at the moment about half of them are in the cadastral stream. The course consists of 3 blocks.

3.2.1 ITC LIS Diploma First Block. The first block covers some fundamentals (including the nature and purpose of GIS in different applications; microcomputer operating systems; wordprocessing; BASIC programming; use of SQL; conceptual database design; data capture verification and storage; data structures for map production; data analysis and spatial modelling processes; georeferencing; geometric transformations; economic role of land; land valuation; legal aspects of cadastre) which are taught to all three streams, and others taught only to specific streams (including airphoto interpretation; geometric transformations; relational databases; computer graphics; FORTRAN; point determination systems; ecology; agronomy; land evaluation).

It can be seen that basic computer science is covered in this block. This reflects the present educational level of most of our course participants, and might be unnecessary for students from industrialised countries. With the advent of low-cost microcomputers students from all countries will soon have these basic computer skills, and the First Block will have to be rethought. At the moment participants who already possess these skills (a tiny minority) can replace this coursework with a personal study topic.

3.2.2 ITC LIS Diploma Second Block. In the Second Block the participants specialise. For the cadastral stream coursework is designed with the objectives of enabling students to compare the appropriateness of one cadastral system to another; to design an efficient system of land registration; to design a computerised land information system in a well known environment (such as Intergraph, Igos, Sysscan, or Arc-Info); and to establish criteria for the effective implementation and management of a cadastral and municipal information system. For the rural and urban streams the objectives are to strengthen the participants' understanding of the analytical processes which have to be applied in geographic problem solving; and to increase their operational familiarity with a large number of mainly micro-computer based GIS.

A rather fundamental difference between the streams emerges in Block 2. To some extent a cadastral LIS is an inventory to be accessed with little data processing needed - thus the prime requirement of the professional is to understand how to DESIGN as good an LIS as possible. For the rural and urban LIS an important task is data analysis and processing - thus the prime requirement of the good professional is to understand how to USE an LIS as well as possible.

3.2.3 ITC LIS Diploma Third Block. Block 3 is devoted to a final project lasting three months. The participants can elaborate the design and implementation of an LIS suitable for use in his home environment. Material from the participant's home country is collected and processed. The software will be documented and can be taken home for further use and elaboration. The block is concluded by the student making a presentation of his final project to staff and participants.

These final projects vary very much from course participant to course participant, but in the cadastral stream (with which the authors are most familiar) emphasis is placed on the participant achieving independence and self-sufficiency in his chosen area. It is most important that participants can go home and begin to implement a GIS if necessary - and because they may be the only professional in their organisation with computer skills, self-sufficiency is an essential. The results are that software must be either completely understood by the participant (e.g. his own) or the software components (e.g. dBASEIII, AutoCad) be completely reliable, and that the hardware components present no maintenance problems (e.g. repairs can be achieved with a screwdriver or mailed parts). Another characteristic of the final projects is that the GIS built by the course participants will use low-cost hardware, unless the participant knows he is going to return to an existing GIS.

3.2.4 ITC LIS Diploma - Other Aspects. Throughout the course visits are made to organisations in several European countries (e.g. Germany, Denmark, U.K., and of course the Netherlands) where LIS are already installed and in use. Also guest speakers from further afield (for example Harry Christie - Canada, Lynn Holstein - Australia, Rebecca Somers - USA) are invited to share their experiences with the participants.

A final aspect of the course which can be mentioned are our 'work-shops'. These are intensive three or four day exercises (additional to those already mentioned), spread throughout the second and third blocks, exposing participants to one particular system at a time. As we have a large and growing number of systems at ITC (Intergraph, IGOS, Sysscan, Arc-Info, ContextVision, Dipix, Gimms, Masmap, Saladin, Cries, dBASEIII, AutoCad, MAP2, SPSS, ILWIS, USEMAP, etc.) which particular systems are used varies from year to year. The cadastral course participants have two such work-shops, the rural four, and the urban eleven.

3.3 GIS Modules within Other Diploma Courses

Our LIS Diploma courses started in 1985, so before GIS was

taught in some other way. It was, and still is - as part of other Diploma courses. In the Cartography department, for example, students have a ten-hour introduction to GIS, and a 40 hour practical exercise in which they build and interrogate a municipal information system. This is of course in addition to about 120 hours in other related digital subjects, and about 70 hours of CAC practical exercises. In the ADAP department it has been an integral part of courses for ten years, similarly other departments.

3.4 Cartography Department in LIS/GIS Teaching at ITC

As map documents are still the most important data source for GIS, Cartographers play a dominant role in teaching all aspects of data capture from maps.

Another function of cartographers in mapping is display design. At the moment cartographers teach display (both hardcopy and softcopy) design to only the Cadastral stream in the ITC LIS Diploma. This anomalous situation arises because the Cadastral stream at ITC is the joint responsibility of the Photogrammetry (ADAP) department and the Cartography department. However Cartography Department research into expert systems guiding good map design is resulting in a map design module for the ITC's own GIS (called ILWIS), and this should open the way for cartographers to influence display design for all the LIS Diploma streams.

4.0 CHARACTERISTICS OF EDUCATION AT ITC

ITC is different from a University in many ways, and these differences affect the way GIS is taught.

The most important difference is our students, or course participants. They are highly motivated mid-career professionals from a very great variety of LDC's. They come to us for only a year, in that year work solidly - without vacations, and in the case of the LIS course participants, expect at the end of the year to have learnt enough to go home and to establish or run (or both) an LIS. Often they are keenly aware of their privileged status and the high expectations of their colleagues at home. This means they may expect to be provided with ready-made GIS solutions or recipes for their problems; instead they are provided with some example solutions to example problems, the means to identify their problems, and some of the intellectual and practical tools with which to build a GIS to solve their problems. In some cases the participants are disappointed by this. Teaching staff are aware of the high expectations of their students and the danger of disappointment. The result is a strong commitment by the staff to ensure that participants do acquire and master the tools forming the bulk of their 'diploma packet' and a very intense involvement by the staff members in the Final Projects, which as already indicated are related to the participant's home situation.

Because of time pressures on our course participants, all learning is expected to be relevant. A standard introductory programming course is not appropriate, and even the earliest programming problems should be designed to strengthen geographic thinking as well as programming skills. Or, as another example, in georeferencing, participants insist that projections used in their own

countries are dealt with.

The result is that much teaching is student driven. The partnership between student and teacher at ITC may be one of ITC's most unique features, and is certainly not found at a University where there is a real age and experience gap between student and teacher.

Another important difference between ITC and a University is that its staff both come from all over the world, and through ITC's consulting arm, work all over the world. Staff have practical experience of the professional domains of our students. The cultural domains of our students are, to a great extent, also known to us so we are aware of which alternative solutions might (given the cultural constraints) help with a particular problem, and which might not. These particular characteristics may not be found in a University staffed by nationals from mainly one country.

We do have students from industrialised countries too. Some are professionals wishing to update among students of similar age and experience, some have chosen to do their thesis work at ITC because of the rather exceptional range of hardware available, while others wish the experience of an international environment.

5.0 THE FUTURE

As with a University, our future is never certain. Changes in the teaching of GIS will arise - especially as our intake becomes increasingly computer literate. At the moment financial pressures ensure the popularity of courses dealing with the cadastral applications of GIS, but as LDC's become aware of the importance of maintaining and improving their natural environment, we may find a re-awakening of interest in satellite remote sensing and the rural applications of GIS.

During the last decade ITC's main thrusts have been in problem-oriented teaching and consulting. Although there has been individual research, strong research groups have not been operating. However two years ago the institute decided to support a group of about 20 staff and students in the creation of a microbased GIS. This is called ILWIS (Integrated Land and Watershed Management Information System), and is particularly suited to rural applications of GIS [VALENZUELA,1988]. Its present status is that it has been established at about ten locations outside Europe, and can handle satellite image enhancement and analysis, map data capture, analysis, and processing, map display and report generation. In its second version it is hoped additional modules will include digital mono-plotting, map design via a cartographic expert system, data and model quality handled by fuzzy sub-set theory, automatic digitizing and vectorizing, etc. ILWIS is the main teaching system for rural GIS applications in ITC, but it will also be the framework for much research at ITC in the next year or two.

Finally, as with any discipline, periods of integration and specialisation alternate in the earth sciences. The emerging GIS technology is a force for integration, and at ITC it is generating integrated research. In the case of teaching, an integrated teaching group has emerged for the

teaching of the ITC LIS Diploma Courses, but the specialist courses will continue - although such students will be expected to fully understand their specialist role within the umbrella technology of GIS.

6.0 REFERENCES

- CHORLEY, Lord R., "Handling Geographic Information - Report to the Secretary of State for the Environment of the Committee of Enquiry into the Handling of Geographic Information", HMSO, London 1987
- JERIE, H.G., KURE, J., LARSEN, H.K., "A systems approach to improving geo-information systems", ITC Journal 1980-4
- LINDEN, G., "Course Calendar 1988-1989 GIS/LIS Courses", ITC 1988
- DE MAN, W.H.E., "A GIS in relation to its use", International Journal of Geographical Information Systems, Vol 2 Nr 3, July-Sept 1988
- VALENZUELA, C.R., "ILWIS Overview", ITC Journal 1988-1

GIS - RELATED EDUCATION AND TRAINING AT SIEMENS

H.J. Vogel
Siemens AG

School for Data Processing and Communication Techniques
P.O. Box 83 09 51, D-8000 Munich 83, FR Germany

ABSTRACT

With its wide range of various software modules SICAD has become a recognized and approved tool for the creation, maintenance, and extension of Geographic Information Systems. SICAD programs may run on microcomputer based graphic workstations in stand-alone mode as well as on large mainframes. The activities and facilities of the company to deliver the necessary education and training to the users of SICAD in order to enable them to master that tool are described. The important role played by the Siemens School for Data Processing and Communication Techniques in this regard is highlighted. The School's offer for professional training and education does not only comprise trainee programmes and courses for customers and Siemens staff. There are also courses held of up to two years that will lead to official qualifications and publicly recognized professional degrees. Efforts are also undertaken in the field of continuing education and re-education and training of the unemployed. The contributions of the company's consultants and site engineers towards education and training of the users by means of individual project support for GIS are mentioned. Finally, a brief outlook is given on the role of user groups, and perspectives on future strategies for education and training are outlined.

INTRODUCTION

Siemens AG is among the World's biggest companies in the fields of electrical engineering and electronics. The company employs more than 320 000 people, the staff of its subsidiaries not included. The network of subsidiaries and branch offices spreads all over the world. One of the company's major activities is the development and manufacturing of a wide variety of computers and their related devices. They are marketed internationally together with the respective system software and numerous packages of applications software, many of them being original in-house developments.

It has always been company policy to offer and deliver far ranging support on the technical side during product installation and thereafter, as well as on the applications side, where experts are available, who are highly proficient in such fields as mechanical, electrical or civil engineering, statistics, surveying or cartography, to name here but a few. If so desired, support can be rendered during every stage of a

user-defined project, i.e. especially during planning, design and implementation. Thus, complete solutions for various complex problems can be offered: hardware and software, networking and personal project support all out of one hand. One such solution is put together under the name of SICAD-Cartography - SICAD standing for Siemens computer aided design. It comprises the full range of tools needed for the layout, implementation and maintenance of Geographical Information Systems (GIS).

In order to be able to realize the described concept of support in a most efficient manner, organized training and education became established practice right from the beginning of the production of computers at Siemens some 30 years ago. Consequently, most of the knowledge how to use SICAD for the creation and maintenance of GIS is transferred to the users by means of courses, hands-on sessions and seminars by the Siemens School for Data Processing and Communication Techniques. In the following, the SICAD-Cartography concept is explained in order to outline its scope and to understand its significance for GIS. Then, after a brief description of the organisation, structure and tasks of the School, it is shown how SICAD training fits in, and how it relates to GIS.

SICAD-CARTOGRAPHY - TOOL KIT FOR GIS

The definitions given for GIS over the years are numerous. A concise and widely accepted one was given by (Bacon, Kanowitz 1986), when they referred to (Tomlinson's 1976) interpretation: Geographic information systems (GIS) are computerized systems designed to store, process and analyse land resource data sets. In this form, land information systems are GIS, too. Looking at the components of GIS makes it clear, why computer manufacturers and software developers should play a role in GIS training and education. According to (Burrough 1986), GIS have three important components:

- computer hardware,
- sets of application software modules, and
- a proper organisational context.

They need to be in balance, if the system is to function satisfactorily. The SICAD-Cartography concept provides the first two components in toto, and far reaching assistance can be rendered for the third one. Hardware and software have a modular structure. SICAD software may run on microcomputers as well as on large mainframes with many terminals. The range of graphics workstations offers a solution for virtually every professional user, pricewise as well. Workstations in the upper range are capable of a combination of vector and raster data processing, they give the choice of more than 16 mio different colours, are equipped with a 3D hardware segment, and can be upgraded by a powerful array processor. All workstations have SINIX processors of 4 Mbyte CPU or more. The smallest station has a 32 Mbyte hard disk, the biggest stand-alone can be upgraded to more than two Gbyte.

SICAD-DIGSY, a digitizing CAD package with special additional functions for surveyors and cartographers, can run on all of them. The operating system SINIX is a UNIX derivative. Image processing and the combination of vector and raster data is done with the help of SICAD-HYGRIS (hybrid graphics information system). The vector software of HYGRIS is the DIGSY package. From DIGSY, an interface exists to the other SICAD software, which is run under the Siemens operating system BS 2000.

For decentralized applications or for starters, the graphics workstation WS 2000 may be used as a stand-alone microcomputer under BS 2000 as well. All BS 2000 SICAD-modules, including the geographical data base and 3D terrain modelling, can run on this workstation in stand-alone mode. The WS 2000 has 8 Mbyte main memory, extendable to 16 and 32 Mbyte. A minimum of 255 Mbyte storage capacity is available on a fixed disk, and up to three disks at 600 Mbyte each may be added. It has its own graphics processor and an IOP of 4 Mbyte main memory. All workstations are available with a standard tablet of 1280x600 mm - 3xA2 - or an A3 tablet. The cross hair cursors have five (standard) or more function keys. Each workstation has a graphics and a b/w alphanumeric screen, where most of the dialog is run. The separate keyboard is available with various national lay-outs for the keys. The graphics screen has a raster refresh rate of 60 Hz, and a resolution of 1280 x 1024 pixels. 256 colours are standard. All sorts of non-exotic plotters may be connected, as well as matrix printers and a tape drive.

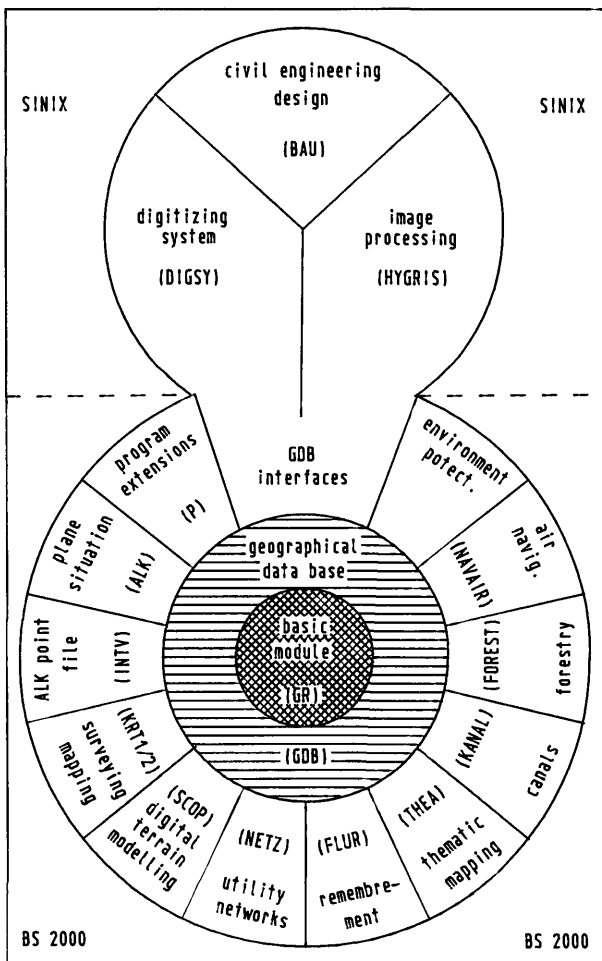
Applications with large area coverage often require outlets at several different sites. Furthermore, it can be an advantage to keep and administrate the data base centrally, e.g. for data security reasons, but have all outlets connected. A mainframe host connected with the graphic terminals via data transfer lines is here the solution. The workstations can have a double function as terminals and stand-alones. The usual transfer rates between 9.6 to 64 kbaud are used.

Whereas DIGSY, HYGRIS and the civil engineering, construction and design package SICAD-BAU are all entities on their own, the BS 2000 packages are all based on the SICAD-GR CAD module. The essential extension package for GIS is the geographic data base (GDB), an integration package that combines a geographical part for continuous maps at high accuracies with a conventional relational type data base for additional non-graphic information that may have a geographical bearing.

The diagram shows that the various software modules can cover a wide range of GIS applications. However, the system is open-ended: a user may write his own program extensions as well as procedures. Except for DIGSY and HYGRIS, which are run in menu mode, all modules are command orientated. A command may be followed by parameters and positions. Command sequences may be

written in procedure form. The user can write his own menus and run the system in menu mode. Five menus with more than 200 menu fields each, where each field may consist of more than 240 characters, can be activated at a time. The system can make use of layer techniques, and can store a geographical area, which may practically be of any size, provided there are enough disks for storage available.

So, SICAD has become a recognized and approved tool for GIS. SICAD systems can be found in many countries. SICAD dialog texts and user manuals are available in English and German. For certain users, text translations have been made into their home languages, e.g. for users in China. Publications about GIS with SICAD and experience gathered date back to the mid-eighties - e.g. (Schilcher 1985), (Vogel 1986) - and have not stopped to appear ever since.



EDUCATION AND TRAINING BY THE SCHOOL

The Siemens School for Data Processing and Communication Techniques is located in Munich, West Germany. Its three main branches are: devices, systems, and professional education and training. Subordinate regional schools are situated in 16 other German towns, well distributed over the entire republic. Siemens subsidiaries in other countries often have their own training facilities, e.g. in Austria, Italy, Portugal, the UK to name here only a few. They are independent from the German School, although they require its services from time to time.

The School's total number of staff amounted to 524 in the past year with more than 330 lecturers, externals not included. The School is open for everybody. The lecturers are experts in their own fields, usually holding BSc, MSc, PhD or similar degrees. Their performance is permanently controlled, e.g. with the help of evaluation sheets each course participant is asked to fill in and hand back by the end of the course. The lecturers themselves undergo ongoing training. Their skills as educators and instructors must be proven, e.g. by qualifying before the examiners board of the public Chamber for Commerce and Industry. 500 different course modules are offered in 150 special rooms for lectures and practicals. 20 main frame computers with 1000 terminals and many PCs are available for those purposes. There were more than 71000 course participants last year, and the average number per course was 17.2. Course fees worth some US \$ 80 mio were received. More than half of all participants' days were spent for courses for professional education and training, re-education and continuing education. Within the branch of professional education, trainee programmes, software development courses, leadership and working methodology, office organisation, expert systems, management seminars, and interactive teachware systems are covered. Many of the topics are manufacturer and product independent. Professional education on behalf of the public labour administration is offered in courses of up to 2 years. Programmers, operators, CAD-designers, CAE systems engineers and others may obtain their professional training and education at the School. More than 2000 candidates make use of that offer at one of the Siemens school sites each year.

The courses are recognized and approved by the public labour administration and are mainly directed towards re-education and continuing education. Many participants are unemployed academics, who can improve their chances for re-employment quite considerably. Such re-education measures are sponsored by the public labour administration. Other high-tec companies have similar schemes in Germany. At the end of the training, examinations have to be passed, and the successful candidates receive publicly recognized certificates about their new professional degrees.

There have also been such courses for people who wished to become CAD cartography designers. They were trained in the necessary background and systems knowledge, the use and application of the operating systems and the use of the hardware. Thereafter, they had to pass all the SICAD course modules in block form. Successful participants can find jobs within Siemens, or at users of GIS.

The compact courses for customers and Siemens staff members as offered by the systems branch of the School, may have even a greater impact on proper GIS training. Three types of courses are held there:

- Single courses for the different modules of SICAD and the acquisition of the necessary knowledge of the operating systems as quoted in the annual course schedule are one type. Target groups, prerequisites for the attendance, course contents and objectives are stated, as are the prices, places and times. 19 different courses are offered in four towns regularly. In a 2 days seminar management and interested parties are informed about the entire range of the SICAD product spectrum. In a full week, an introduction to the BS 2000 and SINIX operating systems is given. For all SICAD software modules single courses, each lasting between 3 and 7 days, are offered. For some modules more than one course is recommended, because of their complexity. Not all courses have to be attended for the various applications, but some 20 to 25 course days spread over a longer or shorter period are recommended, if the user wants to become proficient in the shortest possible time. The price for a 5 day course is less than the equivalent of US \$ 2000.-, tax included.

- The second type are special courses, and block courses in particular. Mainly new SICAD users are addressed. In a very compact form, the participant is introduced into the system environment of SICAD, the use of the operating systems, and essential parts of the SICAD applications software. The block is split in two parts, the first consisting of up to 3 days instruction about the system environment, the use of the operating systems and the graphics workstations, 3 days SICAD-Basic module with the command structure, generation and manipulation commands, image handling, query commands, symbol and menu techniques. Another 2 days follow about the special SICAD procedure technique. SICAD procedure features have great similarity with the variables, arithmetic and logical functions, loops, subroutines and conditions of high level programming languages. Only the very essentials can therefore be taught during these 2 days. 1 to 2 days of instructions about the geographical part of the geographical data base follow. The participant learns how to set up the geographical boundaries for his GIS, how to create the continuous map, and how to achieve data security and define access rights. The first part of the block is concluded by a 4 day long instruction on the special survey and mapping extensions needed for GIS. The generation of special symbols for survey points, slopes etc., the standard survey calculations (polars, joins, area etc.), affine and

Helmert transformations with more than 40 redundant determinations and least squares adjustment, dimensioning, layer techniques, and last but not least the way how to create proper project plans for GIS are being dealt with. This first part comprises a total of 14 working days. As with the other SICAD courses, the mornings are usually reserved for a brief repetition of the previous day's topics, and then followed by theoretical instruction. In the afternoons, there is ample opportunity for all to do practical exercises on the graphics workstations, where usually no more than 3 share one workstation. The second part of the block will last 5 to 10 days, with a 5 day introduction into the handling of the non-graphics part of the geographic data base and the combination with the graphics at the beginning. The other 5 days are set aside for instructions about special applications, e.g. the package about the documentation of utility networks, the 3D digital terrain modelling, or thematic applications - all depending on customer request.

- The third type of courses are those being held on special user request. They usually take place on the user's premises. However, the user must have rooms for lecturing and a sufficient number of graphics workstations must be available on the site.

Block courses and special courses are offered in English and in German. They were held 5 times in 1988. Provided the participants are given sufficient time for practising right afterwards, those courses are the best way to learn as much as possible in the shortest time possible about the use of the tools for the GIS. A lecturer's day is then billed at something less than US \$ 2000.- For big users who start their SICAD GIS, block and special courses are often the most favoured option for getting their staff acquainted and familiarized with the system.

In all SICAD courses reference is made to theoretical fundamentals and practical experience gained in GIS applications. There are also courses being held by university lecturers about general theories behind certain applications. The 5 day SICAD-BILD course about the theories of image processing is one of them. External experts from various fields of applications regularly hold courses and seminars for advanced users. Most of all courses are held on the School's premises in Munich. This is also the place where most of the SICAD program development is done.

User training is not over, once the respective courses have been attended. Siemens site engineers and consultants assist the customer in all stages. They may take part in the project meetings of the various GIS organisation and management groups, give advice in GIS design, organisation, and management aspects, if so desired. If specific know-how for certain side aspects cannot be found within the company, external consultants of renown in their own field, who often work already on a contractor's level for the company, will be called.

Many SICAD users have joined so-called user groups in several countries. They attend meetings held regularly, where experience and information is exchanged. Procedure packages developed by a user are made known to others, and might find interested parties who do not want to re-invent the wheel. Sometimes proposals for certain improvements or extensions of the GIS tools are formulated and submitted to the company. Such proposals are considered in version updates, if they are reasonable and meaningful. Established users sometimes are approached by the company to offer their knowledge gained in practical GIS applications to advanced users in special courses and seminars. Thus, a permanent flow and backfeed of information between supplier and users as well as in between the users themselves is a reality.

CONCLUSION

In order to design, organize and manage a GIS effectively, it is imperative that the user masters the computer hard- and software components excellently. It has been found that intensive course training and continuing support in situ are most effective means to achieve these goals. For comprehensive, large areas covering GIS, a considerable amount of system knowledge - knowledge of the GIS tool kit - is necessary. There are maybe as many different approaches with different software and data structures as there are manufacturers of those systems. Version updates and new releases are still taking place at fast pace. Public institutions concerned with education and training will most probably not be able to invest the amounts necessary to keep abreast with the various developments of all different manufacturers at any time, and provide large area coverage for learners as well. Manufacturers, on the other hand, have the skilled staff to train others from first hand, with lecturers often being involved directly in the respective developments, or having a hot line to the development section of the company. Close cooperation between public institutions concerned with teaching the theories and skills needed for GIS handling and the various manufacturers might become more and more important in the future. Manufacturers cannot and will not take over the role played by independent institutions of the public concerned with teaching - such as universities, technicons, technical high schools and others. But further improvement of the cooperation would certainly be beneficial to users, independent public educational institutions and manufacturers as well.

REFERENCES

Bacon, C.J. & Kanowitz, R.L. 1986, Towards a Geographic Information System for the Siyaya Catchment Project: S.A. Journal of Photogrammetry, Remote Sensing and Cartography, Vol. 14(2), pp. 299-311.

Burrough, P.A. 1986, Principles of Geographical Information Systems for Land Resources Assessment, Clarendon Press, Oxford.

Schilcher, M. 1985, ed. CAD Kartographie, Anwendungen in der Praxis, Herbert Wichmann Verlag, Karlsruhe.

Tomlinson, R.F. et al. 1976, Computer Handling of Geographic Data, Unesco Press, Paris.

Vogel, H.J. 1986, Mapping from digital imagery with an integrated graphics system - Demonstrated at the example of a large town: Allgem. Vermess.-Nachrichten, Internat. Edition, Vol. 3, pp. 3-12.

CARTOGRAPHIC GENERALIZATION IN A DIGITAL ENVIRONMENT: WHEN AND HOW TO GENERALIZE

K. Stuart Shea

The Analytic Sciences Corporation (TASC)
12100 Sunset Hills Road
Reston, Virginia 22090

Robert B. McMaster

Department of Geography
Syracuse University
Syracuse, New York 13244-1160

ABSTRACT

A key aspect of the mapping process—cartographic generalization—plays a vital role in assessing the overall utility of both computer-assisted map production systems and geographic information systems. Within the digital environment, a significant, if not the dominant, control on the graphic output is the role and effect of cartographic generalization. Unfortunately, there exists a paucity of research that addresses digital generalization in a holistic manner, looking at the interrelationships between the conditions that indicate a need for its application, the objectives or goals of the process, as well as the specific spatial and attribute transformations required to effect the changes. Given the necessary conditions for generalization in the digital domain, the display of both vector and raster data is, in part, a direct result of the application of such transformations, of their interactions between one another, and of the specific tolerances required.

How then should cartographic generalization be embodied in a *digital* environment? This paper will address that question by presenting a logical framework of the digital generalization process which includes: a consideration of the intrinsic objectives of **why** we generalize; an assessment of the situations which indicate **when** to generalize; and an understanding of **how** to generalize using spatial and attribute transformations. In a recent publication, the authors examined the first of these three components. This paper focuses on the latter two areas: to examine the underlying conditions or situations when we need to generalize, and the spatial and attribute transformations that are employed to effect the changes.

INTRODUCTION

To fully understand the role that cartographic generalization plays in the digital environment, a comprehensive understanding of the generalization process first becomes necessary. As illustrated in Figure 1, this process includes a consideration of the intrinsic objectives of **why** we generalize, an assessment of the situations which indicate **when** to generalize, and an understanding of **how** to generalize using spatial and attribute transformations. In a recent publication, the authors presented the **why** component of generalization by formulating objectives of the digital generalization process (McMaster and Shea, 1988). The discussion that

follows will focus exclusively on the latter two considerations—an assessment of the degree and type of generalization and an understanding of the primary types of spatial and attribute operations.

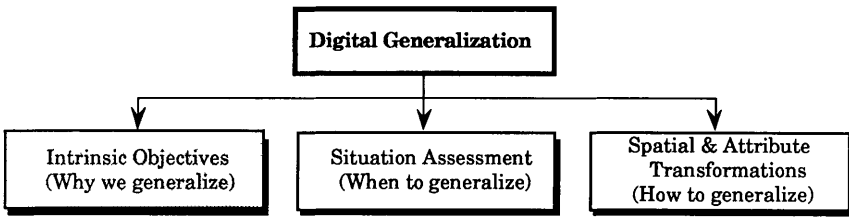


Figure 1. Decomposition of the digital generalization process into three components: **why**, **when**, and **how** we generalize. The **why** component was discussed in a previous paper and will not be covered here.

SITUATION ASSESSMENT IN GENERALIZATION: WHEN TO GENERALIZE

The situations in which generalization would be required ideally arise due to the success or failure of the map product to meet its stated goals; that is, during the cartographic abstraction process, the map fails "...to maintain clarity, with appropriate content, at a given scale, for a chosen map purpose and intended audience" (McMaster and Shea, 1988, p.242). As indicated in Figure 2, the **when** of generalization can be viewed from three vantage points: (1) **conditions** under which generalization procedures would be invoked; (2) **measures** by which that determination was made; and (3) **controls** of the generalization techniques employed to accomplish the change.

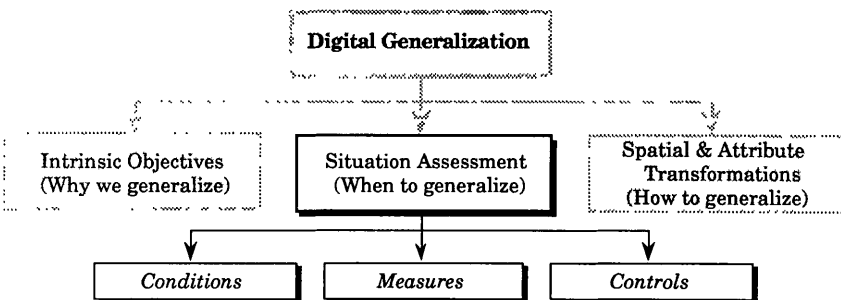


Figure 2. Decomposition of the **when** aspect of the generalization process into three components: *Conditions*, *Measures*, and *Controls*.

Conditions for Generalization

Six conditions that will occur under scale reduction may be used to determine a need for generalization.

Congestion: refers to the problem where too many features have been positioned in a limited geographical space; that is, feature density is too high.

Coalescence: a condition where features will touch as a result of either of two factors: (1) the separating distance is smaller than the resolution of the output device (e.g. pen

width, CRT resolution); or (2) the features will touch as a result of the symbolization process.

Conflict: a situation in which the spatial representation of a feature is in conflict with its background. An example here could be illustrated when a road bisects two portions of an urban park. A conflict could arise during the generalization process if it is necessary to combine the two park segments across the existing road. A situation exists that must be resolved either through symbol alteration, displacement, or deletion.

Complication: relates to an ambiguity in performance of generalization techniques; that is, the results of the generalization are dependent on many factors, for example: complexity of spatial data, selection of iteration technique, and selection of tolerance levels.

Inconsistency: refers to a set of generalization decisions applied non-uniformly across a given map. Here, there would be a bias in the generalization between the mapped elements. Inconsistency is not always an undesirable condition.

Imperceptibility: a situation results when a feature falls below a minimal portrayal size for the map. At this point, the feature must either be deleted, enlarged or exaggerated, or converted in appearance from its present state to that of another—for example, the combination of a set of many point features into a single area feature (Leberl, 1986).

It is the presence of the above stated conditions which requires that some type of generalization process occur to counteract, or eliminate, the undesirable consequences of scale change. The conditions noted, however, are highly subjective in nature and, at best, difficult to quantify. Consider, for example, the problem of congestion. Simply stated, this refers to a condition where the density of features is greater than the available space on the graphic. One might question how this determination is made. Is it something that is computed by an algorithm, or must we rely upon operator intervention? Is it made in the absence or presence of the symbology? Is symbology's influence on *perceived density*—that is, the percent blackness covered by the symbology—the real factor that requires evaluation? What is the unit area that is used in the density calculation? Is this unit area dynamic or fixed? As one can see, even a relatively straightforward term such as density is an enigma. Assessment of the other remaining conditions—coalescence, conflict, complication, inconsistency, and imperceptibility—can also be highly subjective.

How, then, can we begin to assess the state of the condition if the quantification of those conditions is ill-defined? It appears as though such conditions, as expressed above, may be detected by extracting a series of measurements from the original and/or generalized data to determine the presence or absence of a conditional state. These measurements may indeed be quite complicated and inconsistent between various maps or even across scales within a single map type. To eliminate these differences, the assessment of conditions must be based entirely from outside a map product viewpoint. That is, to view the map as a graphic entity in its most elemental form—points, lines, and areas—and to judge the conditions based upon an analysis of those entities. This is accomplished through the evaluation of **measures** which act as indicators into the geometry of individual features, and assess the spatial relationships between combined features. Significant examples of these measures can be found in the cartographic literature (Catlow and Du, 1984; Christ, 1976; Dutton, 1981; McMaster, 1986; Robinson, et al., 1978).

Measures Which Indicate a Need for Generalization

Conditional measures can be assessed by examining some very basic geometric properties of the inter- and intra-feature relationships. Some of these assessments are evaluated in a singular feature sense, others between two independent features, while still others are computed by viewing the interactions of multiple features. Many of these measures are summarized below. Although this list is by no means complete, it does provide a beginning from which to evaluate conditions within the map which do require, or might require, generalization.

Density Measures. These measures are evaluated by using multi-features and can include such benchmarks as the number of point, line, or area features per unit area; average density of point, line, or area features; or the number and location of cluster nuclei of point, line, or area features.

Distribution Measures. These measures assess the overall distribution of the map features. For example, point features may be examined to measure the dispersion, randomness, and clustering (Davis, 1973). Linear features may be assessed by their complexity. An example here could be the calculation of the overall complexity of a stream network (based on say average angular change per inch) to aid in selecting a representative depiction of the network at a reduced scale. Areal features can be compared in terms of their association with a common, but dissimilar area feature.

Length and Sinuosity Measures. These operate on singular linear or areal boundary features. An example here could be the calculation of stream network lengths. Some sample length measures include: total number of coordinates; total length; and the average number of coordinates or standard deviation of coordinates per inch. Sinuosity measures can include: total angular change; average angular change per inch; average angular change per angle; sum of positive or negative angles; total number of positive or negative angles; total number of positive or negative runs; total number of runs; and mean length of runs (McMaster, 1986).

Shape Measures. Shape assessments are useful in the determination of whether an area feature can be represented at its new scale (Christ, 1976). Shape mensuration can be determined against both symbolized and unsymbolized features. Examples include: geometry of point, line, or area features; perimeter of area features; centroid of line or area features; X and Y variances of area features; covariance of X and Y of area features, and the standard deviation of X and Y of area features (Bachi, 1973).

Distance Measures. Between the basic geometric forms—points, lines, and areas—distance calculations can also be evaluated. Distances between each of these forms can be assessed by examining the appropriate shortest perpendicular distance or shortest euclidean distance between each form. In the case of two geometric points, only three different distance calculations exist: (1) point-to-point; (2) point buffer-to-point buffer; and (3) point-to-point buffer. Here, point buffer delineates the region around a point that accounts for the symbology. A similar buffer exists for both line and area features (Dangermond, 1982). These determinations can indicate if any generalization problems exist if, for instance under scale reduction, the features or their respective buffers are in conflict.

Gestalt Measures. The use of Gestalt theory helps to indicate *perceptual* characteristics of the feature distributions through isomorphism—that is, the structural kinship between the stimulus pattern and the expression it conveys (Arnheim, 1974). Common examples of this includes closure, continuation, proximity, similarity, common fate, and figure ground (Wertheimer, 1958).

Abstract Measures. The more *conceptual* evaluations of the spatial distributions can be examined with abstract measures. Possible abstract measures include: homogeneity, neighborliness, symmetry, repetition, recurrence, and complexity.

Many of the above classes of measures can be easily developed for examination in a digital domain, however the Gestalt and Abstract Measures aren't as easily computed. Measurement of the spatial and/or attribute conditions that need to exist before a generalization *action* is taken depends on scale, purpose of the map, and many other factors. In the end, it appears as though many prototype algorithms need first be developed and then tested and fit into the overall framework of a comprehensive generalization processing system. Ultimately, the exact guidelines on how to apply the measures designed above can not be determined without precise knowledge of the algorithms.

Controls on How to Apply Generalization Functionality.

In order to obtain unbiased generalizations, three things need to be determined: (1) the order in which to apply the generalization operators; (2) which algorithms are employed by those operators; and (3) the input parameters required to obtain a given result at a given scale.

An important constituent of the decision-making process is the availability and sophistication of the generalization **operators**, as well as the **algorithms** employed by those operators. The generalization process is accomplished through a variety of these operators—each attacking specific problems—each of which can employ a variety of algorithms. To illustrate, the linear simplification *operator* would access *algorithms* such as those developed by Douglas—as reported by Douglas and Peucker (1973)—and Lang (1969). Concomitantly, there may be permutations, combinations, and iterations of operators, each employing permutations, combinations, and iterations of algorithms. The algorithms may, in turn, be controlled by multiple, maybe even interacting, **parameters**.

Generalization Operator Selection. The control of generalization operators is probably the most difficult process in the entire concept of automating the digital generalization process. These control decisions must be based upon: (1) the importance of the individual features (this is, of course, related to the map purpose and intended audience); (2) the complexity of feature relationships both in an inter- and intra-feature sense; (3) the presence and resulting influence of map clutter on the communicative efficiency of the map; (4) the need to vary generalization amount, type, or order on different features; and (5) the availability and robustness of generalization operators and computer algorithms.

Algorithm Selection. The relative obscurity of complex generalization algorithms, coupled with a limited understanding of the digital generalization process, requires that many of the concepts need to be prototyped, tested, and evaluated against actual requirements. The evaluation process is usually the one that gets ignored or, at best, is only given a cursory review.

Parameter Selection. The input parameter (tolerance) selection most probably results in more variation in the final results than either the generalization operator or algorithm selection as discussed above. Other than some very basic guidelines on the selection of weights for smoothing routines, practically no empirical work exists for other generalization routines.

Current trends in sequential data processing require the establishment of a logical sequence of the generalization process. This is done in order to avoid repetitions of processes and frequent corrections (Morrison, 1975). This sequence is determined by how the generalization processes affect the location and representation of features at the reduced scale. Algorithms required to accomplish these changes should be selected based upon cognitive studies, mathematical evaluation, and design and

implementation trade-offs. Once candidate algorithms exist, they should be assessed in terms of their applicability to specific generalization requirements. Finally, specific applications may require different algorithms depending on the data types, and/or scale.

**SPATIAL AND ATTRIBUTE TRANSFORMATIONS IN GENERALIZATION:
HOW TO GENERALIZE**

The final area of discussion considers the component of the generalization process that actually performs the actions of generalization in support of scale and data reduction. This **how** of generalization is most commonly thought of as the operators which perform generalization, and results from an application of generalization techniques that have either arisen out of the emulation of the manual cartographer, or based solely on more mathematical efforts. Twelve categories of generalization operators exist to effect the required data changes (Figure 3).

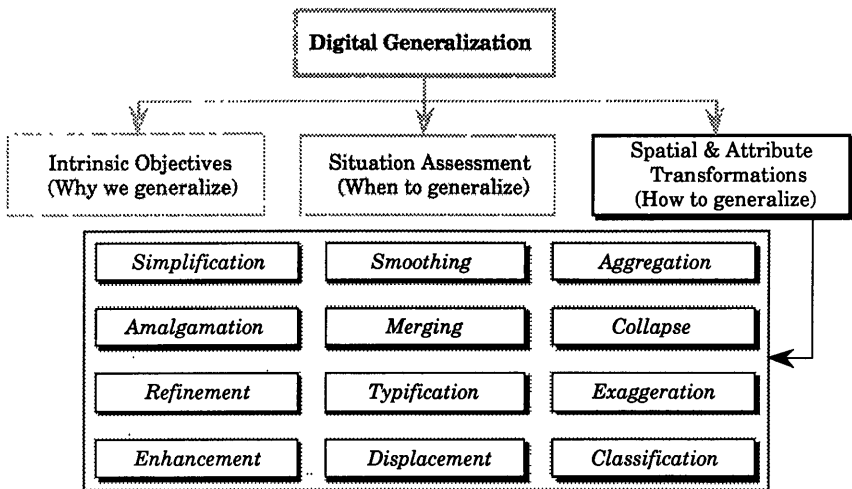


Figure 3. Decomposition of the **how** aspect of the generalization process into twelve operators: simplification, smoothing, aggregation, amalgamation, merging, collapse, refinement, typification, exaggeration, enhancement, displacement, and classification.

Since a map is a reduced representation of the Earth's surface, and as all other phenomena are shown in relation to this, the scale of the resultant map largely determines the amount of information which can be shown. As a result, the generalization of cartographic features to support scale reduction must obviously change the way features look in order to fit them within the constraints of the graphic. Data sources for map production and GIS applications are typically of variable scales, resolution, accuracy—and each of these factors contribute to the method in which cartographic information is presented at map scale. The information that is contained within the graphic has two components—location and meaning—and generalization affects both (Keates, 1973). As the amount of space available for portraying the cartographic information decreases with decreasing scale, less locational information can be given about features, both individually and collectively. As a result, the graphic depiction of the features changes to suit the scale-specific needs. Below, each of these

transformation processes or generalization operators are reviewed. Figure 4 provides a concise graphic depicting examples of each in a format employed by Lichtner (1979).

Simplification. A digitized representation of a map feature should be accurate in its representation of the feature (shape, location, and character), yet also efficient in terms of retaining the least number of data points necessary to represent the character. A profligate density of coordinates captured in the digitization stage should be reduced by selecting a subset of the original coordinate pairs, while retaining those points considered to be most representative of the line (Jenks, 1981). Glitches should also be removed. Simplification operators will select the characteristic, or shape-describing, points to retain, or will reject the redundant point considered to be unnecessary to display the line's character. Simplification operators produce a reduction in the number of derived data points which are unchanged in their x,y coordinate positions. Some practical considerations of simplification includes reduced plotting time, increased line crispness due to higher plotting speeds, reduced storage, less problems in attaining plotter resolution due to scale change, and quicker vector to raster conversion (McMaster, 1987).

Smoothing. These operators act on a line by relocating or shifting coordinate pairs in an attempt to plane away small perturbations and capture only the most significant trends of the line. A result of the application of this process is to reduce the sharp angularity imposed by digitizers (Töpfer and Pillewizer, 1966). Essentially, these operators produce a derived data set which has had a cosmetic modification in order to produce a line with a more aesthetically pleasing caricature. Here, coordinates are shifted from their digitized locations and the digitized line is moved towards the center of the intended line (Brophy, 1972; Gottschalk, 1973; Rhind, 1973).

Aggregation. There are many instances when the number or density of like point features within a region prohibits each from being portrayed and symbolized individually within the graphic. This notwithstanding, from the perspective of the map's purpose, the importance of those features requires that they still be portrayed. To accomplish that goal, the point features must be aggregated into a higher order class feature areas and symbolized as such. For example, if the intervening spaces between houses are smaller than the physical extent of the buildings themselves, the buildings can be aggregated and resymbolized as *built-up areas* (Keates, 1973).

Amalgamation. Through amalgamation of individual features into a larger element, it is often possible to retain the general characteristics of a region despite the scale reduction (Morrison, 1975). To illustrate, an area containing numerous small lakes—each too small to be depicted separately—could with a judicious combination of the areas, retain the original map characteristic. One of the limiting factors of this process is that there is no fixed rule for the degree of detail to be shown at various scales; the end-user must dictate what is of most value. This process is extremely germane to the needs of most mapping applications. Tomlinson and Boyle (1981) term this process *dissolving and merging*.

Merging. If the scale change is substantial, it may be impossible to preserve the character of individual linear features. As such, these linear features must be merged (Nickerson and Freeman, 1986). To illustrate, divided highways are normally represented by two or more adjacent lines, with a separating distance between them. Upon scale reduction, these lines require that they be merged into one positioned approximately halfway between the original two and representative of both.

Collapse. As scale is reduced, many areal features must eventually be symbolized as points or lines. The decomposition of line and area features to point features, or area features to line feature, is a common generalization process. Settlements, airports, rivers, lakes, islands, and buildings, often portrayed as area features on large scale maps, can become point or line features at smaller scales and areal tolerances often guide this transformation (Nickerson and Freeman, 1986).

Refinement. In many cases, where like features are either too numerous or too small to show to scale, no attempt should be made to show all the features. Instead, a selective number and pattern of the symbols are depicted. Generally, this is accomplished by leaving out the smallest features, or those which add little to the general impression of the distribution. Though the overall initial features are thinned out, the general pattern of the features is maintained with those features that are chosen by showing them in their correct locations. Excellent examples of this can be found in the Swiss Society of Cartography (1977). This refinement process retains the general characteristics of the features at a greatly reduced complexity.

Typification. In a similar respect to the refinement process when similar features are either too numerous or too small to show to scale, the typification process uses a representative pattern of the symbols, augmented by an appropriate explanatory note (Lichtner, 1979). Here again the features are thinned out, however in this instance, the general pattern of the features is maintained with the features shown in *approximate* locations.

Exaggeration. The shapes and sizes of features may need to be exaggerated to meet the specific requirements of a map. For example, inlets need to be opened and streams need to be widened if the map must depict important navigational information for shipping. The amplification of environmental features on the map is an important part of the cartographic abstraction process (Muehrcke, 1986). The exaggeration process does tend to lead to features which are in conflict and thereby require displacement (Caldwell, 1984).

Enhancement. The shapes and size of features may need to be exaggerated or emphasized to meet the specific requirements of a map (Leberl, 1986). As compared to the exaggeration operator, enhancement deals primarily with the symbolization component and not with the spatial dimensions of the feature although some spatial enhancements do exist (e.g. fractalization). Proportionate symbols would be unidentifiable at map scale so it is common practice to alter the physical size and shape of these symbols. The delineation of a bridge under an existing road is portrayed as a series of cased lines may represent a feature with a ground distance far greater than actual. This enhancement of the symbology applied is not to exaggerate its meaning, but merely to accommodate the associated symbology.

Displacement. Feature displacement techniques are used to counteract the problems that arise when two or more features are in conflict (either by proximity, overlap, or coincidence). More specifically, the interest here lies in the ability to offset feature locations to allow for the application of symbology (Christ, 1978; Schittenhelm, 1976). The graphic limits of a map make it necessary to move features from what would otherwise be their true planimetric locations. If every feature could realistically be represented at its true scale and location, this displacement would not be necessary. Unfortunately, however, feature boundaries are often an infinitesimal width; when that boundary is represented as a cartographic line, it has a finite width and thereby occupies a finite area on the map surface. These conflicts need to be resolved by: (1) shifting the features from their true locations (displacement); (2) modifying the features (by symbol alteration or interruption); or (3) or deleting them entirely from the graphic.

Classification. One of the principle constituents of the generalization process that is often cited is that of data classification (Muller, 1983; Robinson, et al., 1978). Here, we are concerned with the grouping together of objects into categories of features sharing identical or similar attribution. This process is used for a specific purpose and usually involves the agglomeration of data values placed into groups based upon their numerical proximity to other values along a number array (Dent, 1985). The classification process is often necessary because of the impracticability of symbolizing and mapping each individual value.

Spatial and Attribute Transformations (Generalization Operators)	Representation in the Original Map		Representation in the Generalized Map	
	At Scale of the Original Map		At 50% Scale	
Simplification				
Smoothing				
Aggregation				
Amalgamation				
Merge				
Collapse				
Refinement				
Typification				
Exaggeration				
Enhancement				
Displacement				
Classification	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20	1-5, 6-10, 11-15, 16-20	Not Applicable	

Figure 4. Sample spatial and attribute transformations of cartographic generalization.

SUMMARY

This paper has observed the digital generalization process through a decomposition of its main components. These include a consideration of the intrinsic objectives of **why** we generalize; an assessment of the situations which indicate **when** to generalize, and an understanding of **how** to generalize using spatial and attribute transformations. This paper specifically addressed the latter two components of the generalization process—that is, the **when**, and **how** of generalization—by formulation of a set of assessments which could be developed to indicate a need for, and control the application of, specific generalization operations. A systematic organization of these primitive processes—in the form of operators, algorithms, or tolerances—can help to form a complete approach to digital generalization.

The question of when to generalize was considered in an overall framework that focused on three types of drivers (conditions, measures, and controls). Six conditions (including congestion, coalescence, conflict, complication, inconsistency, and imperceptibility), seven types of measures (density, distribution, length and sinuosity, shape, distance, gestalt, and abstract), and three controls (generalization operator selection, algorithm selection, and parameter selection) were outlined. The application of how to generalize was considered in an overall context that focused on twelve types of operators (simplification, smoothing, aggregation, amalgamation, merging, collapse, refinement, typification, exaggeration, enhancement, displacement, and classification). The ideas presented here, combined with those concepts covered in a previous publication—relating to the first of the three components—effectively serves to detail a sizable measure of the digital generalization process.

REFERENCES

- Arnheim, Rudolf (1974). Art and Visual Perception: A Psychology of the Creative Eye, (Los Angeles, CA: University of California Press).
- Bachi, Roberto (1973), "Geostatistical Analysis of Territories," *Bulletin of the International Statistical Institute*, Proceedings of the 39th session, (Vienna).
- Brophy, D.M. (1972), "Automated Linear Generalization in Thematic Cartography," unpublished Master's Thesis, Department of Geography, University of Wisconsin.
- Caldwell, Douglas R., Steven Zoraster, and Marc Hugus (1984), "Automating Generalization and Displacement Lessons from Manual Methods," *Technical Papers of the 44th Annual Meeting of the ACSM*, 11-16 March, Washington, D.C., 254-263.
- Catlow, D. and D. Du (1984), "The Structuring and Cartographic Generalization of Digital River Data," *Proceedings of the ACSM*, Washington, D.C., 511-520.
- Christ, Fred (1976), "Fully Automated and Semi-Automated Interactive Generalization, Symbolization and Light Drawing of a Small Scale Topographic Map," *Nachrichten aus dem Karten-und Vermessungswesen*, Uhersetzunge, Heft nr. 33:19-36.

- Christ, Fred (1978), "A Program for the Fully Automated Displacement of Point and Line Features in Cartographic Generalizations," *Informations Relative to Cartography and Geodesy*, Translations, 35:5-30.
- Dangermond, Jack (1982), "A Classification of Software Components Commonly Used in Geographic Information Systems," in Peuquet, Donna, and John O'Callaghan, eds. 1983. *Proceedings, United States/Australia Workshop on Design and Implementation of Computer-Based Geographic Information Systems* (Amherst, NY: IGU Commission on Geographical Data Sensing and Processing).
- Davis, John C. (1973), Statistics and Data Analysis in Geology, (New York:John Wiley and Sons), 550p.
- Dent, Borden D. (1985). Principles of Thematic Map Design, (Reading, MA: Addison-Wesley Publishing Company, Inc.).
- Douglas, David H. and Thomas K. Peucker (1973), "Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Character," *The Canadian Cartographer*, 10(2):112-123.
- Dutton, G.H. (1981), "Fractal Enhancement of Cartographic Line Detail," *The American Cartographer*, 8(1):23-40.
- Gottschalk, Hans-Jorg (1973), "The Derivation of a Measure for the Diminished Content of Information of Cartographic Line Smoothed by Means of a Gliding Arithmetic Mean," *Informations Relative to Cartography and Geodesy*, Translations, 30:11-16.
- Jenks, George F. (1981), "Lines, Computers and Human Frailties," *Annals of the Association of American Geographers*, 71(1):1-10.
- Keates, J.S. (1973), Cartographic Design and Production. (New York: John Wiley and Sons).
- Lang, T. (1969), "Rules For the Robot Draughtsmen," *The Geographical Magazine*, 42(1):50-51.
- Leberl, F.W. (1986), "ASTRA - A System for Automated Scale Transition," *Photogrammetric Engineering and Remote Sensing*, 52(2):251-258.
- Lichtner, Werner (1979), "Computer-Assisted Processing of Cartographic Generalization in Topographic Maps," *Geo-Processing*, 1:183-199.
- McMaster, Robert B. (1986), "A Statistical Analysis of Mathematical Measures for Linear Simplification," *The American Cartographer*, 13(2):103-116.
- McMaster, Robert B. (1987), "Automated Line Generalization," *Cartographica*, 24(2):74-111.
- McMaster, Robert B. and K. Stuart Shea (1988), "Cartographic Generalization in a Digital Environment: A Framework for Implementation in a Geographic Information System." *Proceedings, GIS/LIS'88*, San Antonio, TX, November 30—December 2, 1988, Volume 1:240-249.

- Morrison, Joel L. (1975), "Map Generalization: Theory, Practice, and Economics," *Proceedings, Second International Symposium on Computer-Assisted Cartography, AUTO-CARTO II, 21-25 September 1975*, (Washington, D.C.: U.S. Department of Commerce, Bureau of the Census and the ACSM), 99-112.
- Muehrcke, Phillip C. (1986). Map Use: Reading, Analysis, and Interpretation, Second Edition, (Madison: JP Publications).
- Muller, Jean-Claude (1983), "Visual Versus Computerized Seriation: The Implications for Automated Map Generalization," *Proceedings, Sixth International Symposium on Automated Cartography, AUTO-CARTO VI, Ottawa, Canada, 16-21 October 1983* (Ontario: The Steering Committee Sixth International Symposium on Automated Cartography), 277-287.
- Nickerson, Bradford G. and Herbert R. Freeman (1986), "Development of a Rule-based System for Automatic Map Generalization," *Proceedings, Second International Symposium on Spatial Data Handling, Seattle, Washington, July 5-10, 1986*, (Williamsville, NY: International Geographical Union Commission on Geographical Data Sensing and Processing), 537-556.
- Rhind, David W. (1973). "Generalization and Realism Within Automated Cartographic Systems," *The Canadian Cartographer*, 10(1):51-62.
- Robinson, Arthur H., Randall Sale, and Joel L. Morrison. (1978). Elements of Cartography, Fourth Edition, (NY: John Wiley and Sons, Inc.).
- Schittenhelm, R. (1976), "The Problem of Displacement in Cartographic Generalization Attempting a Computer Assisted Solution," *Informations Relative to Cartography and Geodesy, Translations*, 33:65-74.
- Swiss Society of Cartography (1977), "Cartographic Generalization," Cartographic Publication Series, No. 2. English translation by Allan Brown and Arie Kers, ITC Cartography Department, Enschede, Netherlands).
- Tomlinson, R.F. and A.R. Boyle (1981), "The State of Development of Systems for Handling Natural Resources Inventory Data," *Cartographica*, 18(4):65-95.
- Töpfer, F. and W. Pillewizer (1966). "The Principles of Selection, A Means of Cartographic Generalisation," *Cartographic Journal*, 3(1):10-16.
- Wertheimer, M. (1958), "Principles of Perceptual Organization," in Readings in Perception, D. Beardsley and M. Wertheimer, Eds. (Princeton, NJ: Van Nostrand).

CONCEPTUAL BASIS FOR GEOGRAPHIC LINE GENERALIZATION

David M. Mark
National Center for Geographic Information and Analysis
Department of Geography, SUNY at Buffalo
Buffalo NY 14260

BIOGRAPHICAL SKETCH

David M. Mark is a Professor in the Department of Geography, SUNY at Buffalo, where he has taught and conducted research since 1981. He holds a Ph.D. in Geography from Simon Fraser University (1977). Mark is immediate past Chair of the GIS Specialty group of the Association of American Geographers, and is on the editorial boards of *The American Cartographer* and *Geographical Analysis*. He also is a member of the NCGIA Scientific Policy Committee. Mark's current research interests include geographic information systems, analytical cartography, cognitive science, navigation and way-finding, artificial intelligence, and expert systems.

ABSTRACT

Line generalization is an important part of any automated map-making effort. Generalization is sometimes performed to reduce data volume while preserving positional accuracy. However, geographic generalization aims to preserve the recognizability of geographic features of the real world, and their interrelations. This essay discusses geographic generalization at a conceptual level.

INTRODUCTION

The digital cartographic line-processing techniques which commonly go under the term "line generalization" have developed primarily to achieve two practical and distinct purposes: to reduce data volume by eliminating or reducing data redundancy, and to modify geometry so that lines obtained from maps of one scale can be plotted clearly at smaller scales. Brassel and Weibel (in press) have termed these *statistical* and *cartographic* generalization, respectively. Research has been very successful in providing algorithms to achieve the former, and in evaluating them (cf. McMaster, 1986, 1987a, 1987b); however, very little has been achieved in the latter area.

In this essay, it is claimed that Brassel and Weibel's *cartographic* generalization should be renamed *graphical* generalization, and should further be subdivided: *visual* generalization would refer to generalization procedures based on principles of computational

vision, and its principles would apply equally to generalizing a machine part, a cartoon character, a pollen grain outline, or a shoreline. On the other hand, *geographical* generalization would take into account knowledge of the geometric structure of the *geographic feature* or feature-class being generalized, and would be the geographical instance of what might be called *phenomenon-based* generalization. (If visual and geographic generalization do not need to be separated, then a mechanical draftsman, a biological illustrator, and a cartographer all should be able to produce equally good reduced-scale drawings of a shoreline, a complicated machine part, or a flower, irrespectively; such an experiment should be conducted!)

This essay assumes the following:

geographical generalization must incorporate information about the geometric structure of geographic phenomena.

It attempts to provide hints and directions for beginning to develop methods for automated geographical generalization by presenting an overview of some geographic phenomena which are commonly represented by lines on maps. The essay focuses on similarities and differences among geographic features and their underlying phenomena, and on geometric properties which must be taken into account in geographical generalization.

OBJECTIVES OF "LINE GENERALIZATION"

Recently, considerable attention has been paid to theoretical and conceptual principles for cartographic generalization, and for the entire process of map design. This is in part due to the recognition that such principles are a prerequisite to fully-automated systems for map design and map-making. Mark and Buttenfield (1988) discussed over-all design criteria for a cartographic expert system. They divided the map design process into three inter-related components: *generalization*, *symbolization*, and *production*. Generalization was characterized as a process which first *models* geographic phenomena, and then generalizes those models. Generalization was in turn subdivided into: *simplification* (including reduction, selection, and repositioning); *classification* (encompassing aggregation, partitioning, and overlay); and *enhancement* (including smoothing, interpolation, and reconstruction). (For definitions and further discussions, see Mark and Buttenfield, 1988.) Although Mark and Buttenfield's discussion of the modeling phase emphasized a phenomenon-based approach, they did not exclude statistical or other phenomenon-independent approaches. Weibel and Buttenfield

(1988) extended this discussion, providing much detail, and emphasizing the requirements for mapping in a geographic information systems (GIS) environment.

McMaster and Shea (1988) focussed on the generalization process. They organized the top level of their discussion around three questions: Why do we generalize? When do we generalize? How do we generalize? These can be stated more formally as intrinsic objectives, situation assessment, and spatial and attribute transformations, respectively (McMaster and Shea, 1988, p. 241). The rest of their paper concentrated on the first question; this essay will review such issues briefly, but is more concerned with their third objective.

Reduction of Data Volume

Many digital cartographic line-processing procedures have been developed to reduce data volumes. This process has at times been rather aptly termed "line reduction". In many cases, the goal is to eliminate redundant data while changing the geometry of the line as little as possible; this objective is termed "maintaining spatial accuracy" by McMaster and Shea (1988, p. 243). Redundant data commonly occur in cartographic line processing when digital lines are acquired from maps using "stream-mode" digitizing (points sampled at pseudo-constant intervals in x, y, distance, or time); similarly, the initial output from vectorization procedures applied to scan-digitized maps often is even more highly redundant.

One stringent test of a line reduction procedure might be: "can a computer-drafted version of the lines after processing be distinguished visually from the line before processing, or from the line on the original source document?" If the answer to both of these questions is "no", and yet the number of points in the line has been reduced, then the procedure has been successful. A quantitative measure of performance would be to determine the perpendicular distance to the reduced line from each point on the original digital line; for a particular number of points in the reduced line, the lower the root-mean-squared value of these distances, the better is the reduction. Since the performance of the algorithm can often be stated in terms of minimizing some statistical measure of "error", line reduction may be considered to be a kind of "statistical generalization", a term introduced by Brassel and Weibel (in press) to describe minimum-change simplifications of digital elevation surfaces.

Preservation of Visual Appearance and Recognizability

As noted above, Brassel and Weibel (in press) distinguish statistical and cartographic generalization. "Cartographic generalization is used only for graphic display and therefore has to aim at visual effectiveness" (Brassel and Weibel, in press). A process with such an aim can only be evaluated through perceptual testing involving subjects representative of intended map users; few such studies have been conducted, and none (to my knowledge) using generalization procedures designed to preserve visual character rather than merely to simplify geometric form.

Preservation of Geographic Features and Relations

Pannekoek (1962) discussed cartographic generalization as an exercise in applied geography. He repeatedly emphasized that individual cartographic features should not be generalized in isolation or in the abstract. Rather, relations among the geographic features they represent must be established, and then should be preserved during scale reduction. A classic example, presented by Pannekoek, is the case of two roads and a railway running along the floor of a narrow mountain valley. At scales smaller than some threshold, the six lines (lowest contours on each wall of the valley; the two roads; the railway; and the river) cannot all be shown in their true positions without overlapping. If the theme of the maps requires all to be shown, then the other lines should be moved away from the river, in order to provide a distinct graphic image while preserving relative spatial relations (for example, the railway is between a particular road and the river). Pannekoek stressed the importance of showing the transportation lines as being on the valley floor. Thus the contours too must be moved, and higher contours as well; the valley floor must be widened to accommodate other map features (an element of cartographic license disturbing to this budding geomorphometer when J. Ross Mackay assigned the article in a graduate course in 1972!). Nickerson and Freeman (1986) discussed a program that included an element of such an adjustment.

A twisting mountain highway provides another kind of example. Recently, when driving north from San Francisco on highway 1, I was startled by the extreme sinuosity of the highway; maps my two major publishing houses gave little hint, showing the road as almost straight as it ran from just north of the Golden Gate bridge westward to the coast. The twists and turns of the road were too small to show at the map scale, and I have little doubt that positional accuracy was maximized by drawing a fairly straight line following the road's "meander axis". The solution used on some Swiss road maps seems better; winding mountain highways are represented by sinuous lines on the map. Again, I have no doubt that, on a 1:600,000 scale map,

the twists and turns in the cartographic line were of a far higher amplitude than the actual bends, and that the winding road symbols had fairly large positional errors. However, the character of the road is clearly communicated to a driver planning a route through the area. In effect, the road is categorized as a "winding mountain highway", and then represented by a "winding mountain highway symbol", namely a highway symbol drafted with a high sinuosity. Positional accuracy probably was sacrificed in order to communicate geographic character.

A necessary prerequisite to geographic line generalization is the identification of the kind of line, or more correctly, the kind of phenomenon that the line represents (see Buttenfield, 1987). Once this is done, the line may in some cases be subdivided into component elements. Individual elements may be generalized, or replaced by prototypical exemplars of their kinds, or whole assemblages of sub-parts may be replaced by examples of their superordinate class. Thus is a rich area for future research.

GEOGRAPHICAL LINE GENERALIZATION

Geographic phenomena which are represented by lines on topographic and road maps are discussed in this section. (Lines on thematic maps, especially "categorical" or "area-class" boundaries, will almost certainly prove more difficult to model than the more concrete features represented by lines on topographic maps, and are not included in the current discussion.) One important principle is:

many geographic phenomena inherit components of their geometry from features of other kinds.

This seems to have been discussed little if at all in the cartographic literature. Because of these tendencies toward inheritance of geometric structure, the sequence of sub-sections here is not arbitrary, but places the more independent (fundamental) phenomena first, and more derived ones later.

Topographic surfaces (contours)

Principles for describing and explaining the form of the earth's surface are addressed in the science of geomorphology. Geomorphologists have identified a variety of terrain types, based on independent variables such as rock structure, climate, geomorphic process, tectonic effects, and stage of development. Although selected properties of topographic surfaces may be mimicked by statistical surfaces such as fractional Brownian models, a kind of fractal (see Goodchild and Mark 1987 for a review), detailed models

of the geometric character of such surfaces will require the application of knowledge of geomorphology. Brassel and Weibel (in press) clearly make the case that contour lines should never be generalized individually, since they are parts of surfaces; rather, digital elevation models must be constructed, generalized, and then re-contoured to achieve satisfactory results, either statistically or cartographically.

Streams

Geomorphologists divide streams into a number of categories. Channel patterns are either straight, meandering, or braided; there are sub-categories for each of these. Generally, streams run orthogonal to the contours, and on an idealized, smooth, single-valued surface, the stream lines and contours are duals of each other. The statistics of stream planform geometry have received much attention in the earth science literature, especially in the case of meandering channels (see O'Neill, 1987). Again, phenomenon-based knowledge should be used in line generalization procedures; in steep terrain, stream/valley generalization is an intimate part of topographic generalization (cf. Brassel and Weibel, in press).

Shorelines

In a geomorphological sense, shorelines might be considered to "originate" as contours, either submarine or terrestrial. A clear example is a reservoir: the shoreline for a fixed water level is just the contour equivalent to that water level. Any statistical difference between the shoreline of a new reservoir, as drawn on a map, and a nearby contour line on that same map is almost certainly due to different construction methods or to different cartographic generalization procedures used for shorelines and contours. Goodchild's (1982) analysis of lake shores and contours on Random Island, Newfoundland, suggests that, cartographically, shorelines tend to be presented in more detail (that is, are relatively less generalized), while contours on the same maps are smoothed to a greater degree. As sea level changes occur over geologic time, due to either oceanographic or tectonic effects, either there is a relative sea-level rise, in which case a terrestrial contour becomes the new shoreline, or a relative sea-level fall, to expose a submarine contour as the shoreline.

Immediately upon the establishment of a water level, coastal geomorphic processes begin to act on the resulting shoreline; the speed of erosion depends on the shore materials, and on the wave, wind, and tidal environment. It is clear that coastal geomorphic processes are scale-dependent, and that the temporal and spatial scales of such processes are functionally linked. Wave refraction

tends to concentrate wave energy at headlands (convexities of the land), whereas energy per unit length of shoreline is below average in bays. Thus, net erosion tends to take place at headlands, whereas net deposition occurs in the bays. On an irregular shoreline, beaches and mudflats (areas of deposition) are found largely in the bays. The net effect of all this is that shorelines tend to straighten out over time. The effect will be evident most quickly at short spatial scales.

Geomorphologists have divided shorelines into a number of types or classes. Each of these types has a particular history and stage, and is composed of members from a discrete set of coastal landforms. Beaches, rocky headlands, and spits are important components. Most headlands which are erosional remnants are rugged, have rough or irregular shorelines, and otherwise have arbitrary shapes determined by initial forms, rock types and structures, wave directions, *et cetera*. Spits and beaches, however, have forms with a much more controlled (less variable) geometry. For example, the late Robert Packer of the University of Western Ontario found that many spits are closely approximated by logarithmic spirals (Packer, 1980).

Political and Land Survey Boundaries

Most political boundaries follow either physical features or lines of latitude or longitude. Both drainage divides (for example, the France-Spain border in the Pyrenees, or southern part of the British Columbia-Alberta in the Rocky Mountains) and streams (there are a great many many examples) are commonly used as boundaries. The fact that many rivers are dynamic in their planform geometry leads to interesting legal and/or cartographic problems. For example, the boundary between Mississippi and Louisiana is the midline of the Mississippi River when the border was legally established more than a century ago, and does not correspond with the current position of the river.

In areas which were surveyed before they were settled by Europeans, rectangular land survey is common. Then, survey boundaries may also be used as boundaries for minor or major political units. Arbitrary lines of latitude or longitude also often became boundaries as a result of negotiations between distant colonial powers, or between those powers and newly-independent former colonies. An example is the Canada - United States boundary in the west, which approximates the 49th parallel of latitude from Lake-of-the-Woods to the Pacific. Many state boundaries in the western United States are the result of the subdivision of larger territories by officials in Washington. Land survey boundaries are rather "organic" and irregular in the metes-and-bounds systems of most of the original 13 colonies of the United States, and in many other parts of the world. They are often much more rectangular in

the western United States, western Canada, Australia, and other "pre-surveyed" regions.

Roads

Most roads are constructed according to highway engineering codes, which limit the tightness of curves for roads of certain classes and speeds. These engineering requirements place smoothness constraints on the short-scale geometry of the roads; these constraints are especially evident on freeways and other high-speed roads, and should be determinable from the road type, which is included in the USGS DLG feature codes and other digital cartographic data schemes. However, the longer-scale geometry of these same roads is governed by quite different factors, and often is inherited from other geographic features.

Some roads are "organic", simply wandering across country, or perhaps following older walking, cattle, or game trails. However, many roads follow other types of geographic lines. Some roads "follow" rivers, and others "follow" shorelines. In the United States, Canada, Australia, and perhaps other countries which were surveyed before European settlement, many roads follow the survey lines; in the western United States and Canada, this amounts to a 1 by 1 mile grid (1.6 by 1.6 km) of section boundaries, some or all of which may have actual roads along them. Later, a high-speed, limited access highway may minimize land acquisition costs by following the older, survey-based roadways where practical, with transition segments where needed to provide sufficient smoothness (for example, Highway 401 in south-western Ontario).

A mountain highway also is an example of a road which often follows a geographic line, most of the time. In attempting to climb as quickly as possible, subject to a gradient constraint, the road crosses contours at a slight angle which can be calculated from the ratio of the road slope to the hill slope. [The sine of the angle of intersection (on the map) between the contour and the road is equal to the ratio of the road slope to the hill slope, where both slopes are expressed as tangents (gradients or percentages).] Whenever the steepness of the hill slope is much greater than the maximum allowable road gradient, most parts of the trace of the road will have a very similar longer-scale geometry to a contour line on that slope. Of course, on many mountain highways, such sections are connected by short, tightly-curved connectors of about 180 degrees of arc, when there is a "switch-back", and the hillside switches from the left to the right side of the road (or the opposite).

Railways

Railways have an even more constrained geometry than roads, since tight bends are never constructed, and gradients must be very low. Such smoothness should be preserved during generalization, even if curves must be exaggerated in order to achieve this.

Summary

The purpose of this essay has not been to criticize past and current research on computerized cartographic line generalization. Nor has it been an attempt to define how research in this area should be conducted in the future. Rather, it has been an attempt to move one (small) step toward a truly "geographic" approach to line generalization for mapping. It is a bold assertion on my part to state that, in order to successfully generalize a cartographic line, one must take into account the geometric nature of the real-world phenomenon which that cartographic line represents, but nevertheless I assert just that. My main purpose here is to foster research to achieve that end, or to debate on the validity or utility of my assertions.

Acknowledgements

I wish to thank Bob McMaster for the discussions in Sydney that convinced me that it was time for me to write this essay, Babs Buttenfield for many discussions of this material over recent years, and Rob Weibel and Mark Monmonier for their comments on earlier drafts of the material presented here; the fact that each of them would dispute parts of this essay does not diminish my gratitude to them. The essay was written partly as a contribution to Research Initiative #3 of the National Center for Geographic Information and Analysis, supported by a grant from the National Science Foundation (SES-88-10917); support by NSF is gratefully acknowledged. Parts of the essay were written while Mark was a Visiting Scientist with the CSIRO Centre for Spatial Information Systems, Canberra, Australia.

References

- Brassel, K. E., and Weibel, R., in press. A review and framework of automated map generalization. *International Journal of Geographical Information Systems*, forthcoming.
- Buttenfield, B. P., 1987. Automating the identification of cartographic lines. *The American Cartographer* 14: 7-20.

- Goodchild, M. F., 1982. The fractional Brownian process as a terrain simulation model. *Modeling and Simulation* 13: 1122-1137. Proceedings, 13th Annual Pittsburgh Conference on Modeling and Simulation.
- Goodchild, M. F., and Mark, D. M., 1987. The fractal nature of geographic phenomena. *Annals of the Association of American Geographers* 77: 265-278.
- Mandelbrot, B. B., 1967. How long is the coast of Britain? Statistical self-similarity and fractional dimension. *Science* 156: 636-638.
- Mark, D. M., and Buttenfield, B. P., 1988. Design criteria for a cartographic expert system. Proceedings, 8th International Workshop on Expert Systems and Their Applications, vol. 2, pp. 413-425.
- McMaster, R. B., 1986. A statistical analysis of mathematical measures for linear simplification. *The American Cartographer* 13: 103-116.
- McMaster, R. B., 1987a. Automated line generalization. *Cartographica* 24: 74-111.
- McMaster, R. B., 1987b. The geometric properties of numerical generalization. *Geographical Analysis* 19: 330-346.
- McMaster, R. B., and Shea, K. S., 1988. Cartographic generalization in digital a environment: A framework for implementation in a geographic information system. *Proceedings, GIS/LIS '88*, vol. 1, pp. 240-249.
- O'Neill, M. P., 1987. Meandering channel patterns-- analysis and interpretation. Unpublished PhD dissertation, State University of New York at Buffalo.
- Packer, R. W., 1980. The logarithmic spiral and the shape of drumlins. Paper presented at the Joint Meeting of the Canadian Association of Geographers, Ontario Division, and the East Lakes Division of the Association of American Geographers, London, Ontario, November 1980.
- Pannekoek, A. J., 1962. Generalization of coastlines and contours. *International Yearbook of Cartography* 2: 55-74.
- Weibel, R., and Buttenfield, B. P., 1988. Map design for geographic information systems. *Proceedings, GIS/LIS '88*, vol. 1, pp. 350-359.

DATA COMPRESSION AND CRITICAL POINTS DETECTION USING NORMALIZED SYMMETRIC SCATTERED MATRIX

Khagendra Thapa B.Sc. B.Sc(Hons) CNAA, M.Sc.E. M.S. Ph.D.
Department of Surveying and Mapping Ferris State University
Big Rapids, Michigan 49307

BIOGRAPHICAL SKETCH

Khagendra Thapa is an Associate Professor of Surveying and Mapping at Ferris State University. He received his B.Sc. in Mathematics, Physics, and Statistics from Tribhuvan University Kathmandu, Nepal and B.Sc.(Hons.) CNAA in Land Surveying from North East London Polytechnic, England, M.Sc.E. in Surveying Engineering from University of New Brunswick, Canada and M.S. and Ph.D. in Geodetic Science from The Ohio State University. He was a lecturer at the Institute of Engineering, Kathmandu Nepal for two years. He also held various teaching and research associate positions both at The Ohio State University and University of New Brunswick.

ABSTRACT

The problems of critical points detection and data compression are very important in computer assisted cartography. In addition, the critical points detection is very useful not only in the field of cartography but in computer vision, image processing, pattern recognition, and artificial intelligence. Consequently, there are many algorithms available to solve this problem but none of them are considered to be satisfactory. In this paper, a new method of finding critical points in digitized curve is explained. This technique, based on the normalized symmetric scattered matrix is good for both critical points detection and data compression. In addition, the critical points detected by this algorithm are compared with those detected by humans.

INTRODUCTION

The advent of computers have had a great impact on mapping sciences in general and cartography in particular. Now-a-days-more and more existing maps are being digitized and attempts have been made to make maps automatically using computers. Moreover, once we have the map data in digital form we can make maps for different purposes very quickly and easily. Usually, the digitizers tend to digitize more data than what is required to adequately represent the feature. Therefore, there is a need for data compression without destroying the character of the feature. This can be achieved by the process of critical points detection in the digital data. There are many algorithms available in the literature for the purpose of critical points detection. In this paper, a new method of critical points detection is described which is efficient and has a sound theoretical basis as it uses the eigenvalues of the Normalized Symmetric Scattered (NSS) matrix derived from the digitized data.

DEFINITION OF CRITICAL POINTS

Before defining the critical points, it should be noted that critical points in a digitized curve are of interest not only in the field of cartography but also in other disciplines such as Pattern Recognition, Image Processing, Computer Vision, and Computer Graphics. Marino (1979) defined critical points as "Those points which remain more or less fixed in position, resembling a precis of the written essay, capture the nature or character of the line".

In Cartography one wants to select the critical points along a digitized line so that one can retain the basic character of the line. Researchers both in the field of Computer Vision and Psychology have claimed that the maxima, minima and zeroes of curvature are sufficient

to preserve the character of a line. In the field of Psychology, Attneave (1954) demonstrated with a sketch of a cat that the maxima of curvature points are all one needs to recognize a known object. Hoffman and Richards (1982) suggested that curves should be segmented at points of minimum curvature. In other words, points of minimum curvature are the critical points. They also provided experimental evidence that humans segmented curves at points of curvature minima. Because the minima and maxima of a curve depend on the orientation of the curve, the following points are considered as critical points:

1. curvature maxima
2. curvature minima
3. end points
4. points of intersection.

It should be noted that Freeman (1978) also includes the above points in his definition of critical points.

Hoffman and Richards (1982) state that critical points found by first finding the maxima, minima, and zeroes of curvature are invariant under rotations, translations, and uniform scaling. Marimont (1984) has experimentally proved that critical points remain stable under orthographic projection.

The use of critical points in the fields of Pattern Recognition, and Image Processing has been suggested by Brady (1982), and Duda and Hart (1973). The same was proposed for Cartography by Solovitskiy (1974), Marino (1978), McMaster (1983), and White (1985).

Importance of Critical Point Detection in Line Generalization

Cartographic line generalization has hitherto been a subjective process. When one wants to automate a process which has been vague and subjective, many difficulties are bound to surface. Such is the situation with Cartographic line generalization. One way to tackle this problem would be to determine if one can quantify it (i.e. make it objective) so that it can be solved using a digital computer. Many researchers such as Solovitskiy (1974), Marino (1979), and White (1985) agree that one way to make the process of line generalization more objective is to find out what Cartographers do when they perform line generalization by hand? In addition, find out what in particular makes the map lines more informative to the map readers. Find out if there is anything in common between the map readers and map makers regarding the interpretation of line character.

Marino (1979) carried out an empirical experiment to find if Cartographers and non-cartographers pick up the same critical points from a line. In the experiment, she took different naturally occurring lines representing various features. These lines were given to a group of Cartographers and a group of non-cartographers who were asked to select a set of points which they consider to be important to retain the character of the line. The number of points to be selected was fixed so that the effect of three successive levels or degrees of generalization could be detected. She performed statistical analysis on the data and found that cartographers and non-cartographers were in close agreement as to which points along a line must be retained so as to preserve the character of these lines at different levels of generalization.

When one says one wants to retain the character of a line what he/she really means is that he/she wants to preserve the basic shape of the line as the scale of representation decreases. The purpose behind the retention of the basic shape of the line is that the line is still recognized as a particular feature- river, coastline or boundary despite of the change in scale. The assumption behind this is that the character of different types of line is different. That is to say that the character of a coastline is different from that of a road. Similarly, the character of a river would be different from that of a transmission line and so on.

The fact that during the process of manual generalization one retains the basic shape of the feature has been stated by various veteran Cartographers. For example, Keates (1973) states, "... each individual feature has to be simplified in form by omitting minor irregularities and retaining only the major elements of the shape". Solovitskiy (1974) identified the following quantitative and qualitative criteria for a correct generalization of lines:

1. The quantitative characteristics of a selection of fine details of a line.
2. Preservation of typical tip angles and corners
3. Preservation of the precise location of the basic landform lines.
4. Preservation of certain characteristic points.
5. Preservation of the alternation frequency and specific details.

He further states "The most important qualitative criteria are the preservation of the general character of the curvature of a line, characteristic angles, and corners...". In the above list, what Solovitskiy is basically trying to convey is that he wants to retain the character of a feature by preserving the critical points. Buttenfield (1985) also points out the fact that Cartographers try to retain the basic character of a line during generalization. She states "... Cartographer's attempt to cope objectively with a basically inductive task, namely, retaining the character of a geographic feature as it is represented at various Cartographic reductions".

Boyle (1970) suggested that one should retain the points which are more important (i.e. critical points) during the process of line generalization. He further suggested that these points should be hierarchical and should be assigned weights (1-5) to help Cartographers decide which points to retain.

Campbell (1984) also observes the importance of retaining critical features. He states, "One means of generalization involves simply selecting and retaining the most critical features in a map and eliminating the less critical ones". The fact that retention of shape is important in line generalization is also included in the definition of line generalization. The DMA (Defense Mapping Agency) definition states as "Smoothing the character of features without destroying their visible shape". Tobler as referenced in Steward (1974) also claims that the prime function of generalization is "... to capture the essential characteristics of ... a class of objects, and preserve these characteristics during the change of scale".

Advantages of Critical Point Detection

According to Pavlidis and Horowitz (1974), Roberge (1984), and McMaster (1983) the

detection and retention of critical points in a digital curve has the following advantages:

1. Data compaction as a result plotting or display time will be reduced and less storage will be required.
2. Feature extraction.
3. Noise filtering.
4. Problems in plotter resolution due to scale change will be avoided.
5. Quicker vector to raster conversion and vice-versa.
6. Faster choropleth shading. This means shading color painting the polygons.

Because of the above advantages, research in this area is going on in various disciplines such as Computer Science, Electrical Engineering, Image Processing, and Cartography.

Literature Review

The proliferation of computers not only has had a great impact on existing fields of studies but also created new disciplines such as Computer Graphics, Computer Vision, Pattern Recognition, Image Processing, Robotics etc. Computers play an ever increasing role in modern day automation in many areas. Like many other disciplines, Mapping Sciences in general and Cartography in particular have been greatly changed due to the use of computers. It is known from experience that more than 80% of a map consists of lines. Therefore, when one talks about processing maps, one is essentially referring to processing lines. Fortunately, many other disciplines such as Image Processing, Computer Graphics, and Pattern Recognition are also concerned with line processing. They might be interested in recognizing shapes of various objects, industrial parts recognition, feature extraction or electrocardiogram analysis etc.

Whatever may be the objective of line processing and whichever field it may be, there is one thing in common viz: it is necessary to retain the basic character of the line under consideration. As mentioned above one needs to detect and retain the critical points in order to retain the character of a line. There is a lot of research being carried out in all the above disciplines as to the detection of critical points. Because the problem of critical points detection is common to so many disciplines, it has many nomenclatures. A number of these nomenclatures (Wall and Danielson, 1984), (Dunham, 1986), (Imai and Iri, 1986), (Anderson and Bezdek, 1984), (Herkommer, 1985), (Freeman and Davis, 1977), (Rosenfeld and Johnston, 1973), (Rosenfeld and Thurston, 1971), (Duda and Hart, 1973), (Opheim, 1982), (Williams, 1980), (Roberge, 1984), (Pavlidis and Horowitz, 1974), (Fischler and Bolles, 1983,1986), (Dettori and Falcidieno, 1982), (Reumann and Witkam, 1974), (Sklansky and Gonzlax, 1980), (Sharma and Shanker, 1978), (Williams, 1978) are listed below:

1. Planer curve segmentation
2. Polygonal Approximation
3. Vertex Detection
4. Piecewise linear approximation
5. Corner finding
6. Angle detection
7. Line description

8. Curve partitioning
9. Data compaction
10. Straight line approximation
11. Selection of main points
12. Detection of dominant points
13. Determination of main points.

Both the amount of literature available for the solution of this problem and its varying nomenclature indicate the intensity of the research being carried out to solve this problem. It is recognized by various researchers (e.g. Fischler and Bolles, 1986) that the problem of critical points detection is in fact a very difficult one and it still is an open problem. Similarly, the problem of line generalization is not very difficult if carried out manually but becomes difficult if one wants to do it by computer. Because of the subjective nature of this problem and due to the lack of any criteria for evaluation of line generalization, it has been very difficult to automate this process. Recently some researchers for example (Marino, 1979) and (White, 1985) have suggested that one should first find critical points and retain them in the process of line generalization.

Algorithms for Finding Critical Points

As noted in the previous section, there are many papers published on critical points detection which is identified by different names by different people. It should, however, be noted that the detection is not generic but, as indicated by Fischler and Bolles (1986) depends on the following factors:

1. purpose
2. vocabulary
3. data representation
4. past experience of the 'partitioning instrument'. In cartography it would mean the past experience of the cartographer.

It is interesting to note that the above four factors are similar to the controls of line generalization that Robinson et al. (1985) have pointed out. However, the fourth factor viz: past experience and mental stability of the Cartographer is missing from the latter list.

THE NATURE OF SCATTER MATRICES AND THEIR EIGENVALUES

Consider the geometry of the quadratic form associated with a sample covariance matrix. suppose $P = (p_1, p_2, \dots, p_n)$ be a finite data set in R^2 and P is a sample of n independently and identically distributed observations drawn from real two dimensional population.

Let (μ, Σ) denote the population mean vector and variance matrix and let (v_p, V_p) be the corresponding sample mean vector and sample covariance matrix these are then given by (Uotila, 1986)

$$v_p = \sum p_i/n; \quad V_p = \sum (p_i - v_p)(p_i - v_p) \quad (1)$$

Multiply both sides of the equation for V_p by $(n - 1)$ and denote the RHS by S_p viz:

The matrices S_p and V_p are both 2×2 symmetric and positive semi-definite. Since these matrices are multiples of each other they share identical eigen-spaces.

According to Anderson and Bezdek (1983) one can use the eigenvalue and eigenvector structure of S_p to extract the shape information of the data set it represents. This is because the shape of the data set is supposed to mimic the level shape of the probability density function $f(x)$ of x . For example, if the data set is bivariate normal, S_p has two real, non-negative eigenvalues. Let these eigenvalues be λ_1 and λ_2 . Then the following possibilities exist (Anderson and Bezdek, 1983):

1. If both λ_1 and $\lambda_2 = 0$, then the data set P is degenerate, and S_p is invertible and there exist with probability 1, constants a , b , and c such the $ax + by + c = 0$. In this case the sample data in P lie on a straight line.
2. If $\lambda_1 > \lambda_2 > 0$, then the data set represent an elliptical shape.
3. If $\lambda_1 = \lambda_2 > 0$, then the sample data set in P represent a circle.

EIGENVALUES OF THE NORMALIZED SYMMETRIC SCATTER MATRIX (NSS)

Supposing that one has the following data:

$$P = (P_1, P_2, \dots, P_n) \\ \text{where } P_i = (x_i, y_i)$$

Then the normalized scattered matrix A is defined as

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = S_p / \text{trace}(S_p) \quad (3)$$

For the above data set A is given by:

$$\text{Deno} = \sum ((x_i - x_m)^2 + (y_i - y_m)^2) \quad (4)$$

$$a_{11} = 1/\text{Deno} \sum (x_i - x_m)^2$$

$$a_{12} = 1/\text{Deno} \sum (x_i - x_m)(y_i - y_m)$$

$$a_{21} = 1/\text{Deno} \sum (x_i - x_m)(y_i - y_m) \quad (5)$$

$$a_{22} = 1/\text{Deno} \sum (y_i - y_m)^2$$

where $v_x = (x_m, y_m)$ is the mean vector defined as

$$x_m = \sum x_i / n, \text{ and } y_m = \sum y_i / n \quad (6)$$

Note that the denominator in (3) will vanish only when all the points under consideration are identical.

The characteristic equation of A is given by:

$$|A - \lambda I| = 0 \quad (7)$$

which may be written as (for 2×2 matrix)

$$|A - \lambda I| = 0 \quad (7)$$

which may be written as (for 2x2 matrix)

$$\lambda^2 - \text{trace}(A)\lambda + \text{Det}(A) = 0 \quad (8)$$

where $\text{Det}(A)$ = Determinant of A.

By design the trace of A is equal to 1. Hence the characteristics equation of A reduces to

$$\lambda^2 + \text{Det}(A) = 0 \quad (9)$$

The roots of this equation are the eigenvalues and are given by:

$$\lambda_1 = (1 + \sqrt{1 - 4*\text{Det}(A)})/2 \text{ and } \lambda_2 = (1 - \sqrt{1 - 4*\text{Det}(A)})/2 \quad (10)$$

For convenience put $D_x = \sqrt{1 - 4*\text{Det}(A)}$, then

$$\lambda_1 = (1 + D_x)/2 \quad (11)$$

$$\lambda_2 = (1 - D_x)/2 \quad (12)$$

Now λ_1 and λ_2 satisfy the following two conditions:

$$\lambda_1 + \lambda_2 = 1 \quad (13)$$

Since the sum of the roots of an equation of the form

$$ax^2 + bx + c = 0 \text{ are } \lambda_1 + \lambda_2 = -b/a$$

Subtracting (12) from (11), one obtains

$$\lambda_1 - \lambda_2 = D_x \quad (14)$$

Since the eigenvalues λ_1 and λ_2 satisfy the equations (13) and (14) the three cases discussed previously reduce to the following from (Anderson and Bezdek, 1983):

1. The data set represent a straight line if and only if $D_x = 1$
2. The data set represent an elliptical shape if and only if $0 < D_x < 1$
3. The data set represent a circular shape if $D_x = 0$.

ALGORITHM TO DETECT CRITICAL POINTS USING NSS MATRIX

The fact that the analysis of the eigenvalues of the NSS matrix can be used to extract shape of the curve represented by the data set, may be exploited to detect critical points in the digital curve. Assuming that the data is gross error free, and devoid of excessive noise, one can outline the algorithm to detect critical points in the following steps:

3. If D_x is greater than a certain tolerance (e.g. 0.95) add one more point to the data and repeat from step 2.
4. If D_x is less than the tolerance point, point 2 is a critical point. Retain point 2 and repeat the process from step 1 with point two as the new starting point.
5. Repeat the process until the end of the data set is reached.

Results of Critical Points Detection by NSS Matrix

The algorithm discussed in the previous section is useful in detecting the critical points in vector data. The only parameter involved in this technique is D_x which was defined earlier. by varying the value of D_x between say 0.8 to 1.0 one can get a varying amount of detail in a curve. Figure 1 shows the selected critical points for the test figure for $D_x = 0.96$.

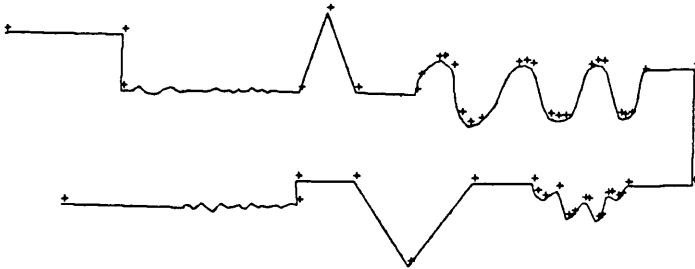


Figure 1: Results of Critical Points Selection by NSS Matrix. There were 50 points selected.

There are 50 points selected in this figure. It is clear from the figure that this method will be very useful for compression of digitized data since it retains the overall shape of the curve without retaining the unnecessary points.

COMPARISON BETWEEN MANUAL AND ALGORITHMIC CRITICAL POINTS DETECTION

In this section, results of critical points detection in the test figure by a group of people are given. These results are then compared with the results obtained from the NSS matrix technique of critical points detection.

MANUAL CRITICAL POINTS DETECTION: THE EXPERIMENT

In order to find if the NSS matrix method critical points detection can mimic humans or not, the test figure was given to a group of 25 people who had at least one course in Cartography.

REFERENCES

- Anderson, I.M. and J.C. Bezdek (1984), "Curvature and Tangential Deflection of Discrete Arcs: A Theory Based on the Commutator of Scatter Matrix Pairs and Its Application to Vertex Detection in Planer Shape Data", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, NO. 1, pp. 27-40.
- Attneave, F. (1954), "Some Informational Aspects of Visual Perception", Psychological Review, Vol. 61, pp. 183-193.
- Boyle, A.R. (1970), "The Quantized Line", The Cartographic Journal, Vol. 7, No. 2, pp. 91-94.
- Buttenfield, B. (1985), "Treatment of the Cartographic Line", Cartographica, Vol. 22, No. 2, pp. 1-26.
- Campbell, J. (1984), Introductory Cartography, Prentice Hall, Inc., Englewood Cliffs, NJ 07632.
- Davis, L.S. (1977), "Understanding Shape: Angles and Sides", IEEE Transactions on Computers, Vol. C-26, No.3, pp. 236-242.
- Dettori, G. and B. Falcidieno (1982), "An Algorithm for Selecting Main Points on a Line," Computers and Geosciences, Vol. 8, pp.3-10.
- Douglas, D.H. and T. K. Peucker (1973), "Algorithms for the Reduction of the Number of points Required to Represent a Digitized Line for its Character", The Canadian Cartographer Vol. 10.
- Duda, R.O. and P.E. Hart (1973), Pattern Classification and Scene Analysis, Willey Interscience.
- Dunham, J.G. (1986), "Optimum Uniform Piecewise Linear Approximation of Planer Curves", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 1.
- Fischler, M.A. and R.C. Bolles (1986), "Perceptual Organization and Curve Partitioning," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol PAMI-8, No. 1.
- Freeman, H. (1978), "Shape Description Via the Use of Critical Points," Pattern Recognition, Vol. 10, pp. 159-166.
- Herkommer, M.A. (1985), "Data-Volume Reduction of Data Gathered along Lines Using the Correlation Coefficient to Determine Breakpoints," Computers and Geosciences, Vol. 11, No. 2, pp. 103-110.

- Hoffman, D.D. and W.A. Richards (1982), "Representing Smooth Plane Curves for Recognition: Implications for Figure-Ground Reversal," Proceedings of the National Conference on Artificial Intelligence," Pittsburgh PA, pp. 5-8.
- Imai, H. and M. Iri (1986), "Computational Geometric Methods for Polygonal Approximations of a Curve," Computer Vision Graphics and Image Processing, Vol. 36, pp. 31-41.
- Keates, J.S. (1981), Cartographic Design and Production, Thetford, Great Britain: Longman.
- Marino, J. S. (1979), "Identification of Characteristics Points along Naturally Occuring Lines: An Empirical Study," The Canadian Cartographer, Vol. 16, No. 1, pp. 70-80.
- Marino, J. (1978), "Characteristics Points and Their Significance to Line Generalization," Unpublished M.A. Thesis University of Kansas.
- McMaster, R.B. (1983), "A Quantative Analysis of Mathematical Measures in Linear Simplification," Ph.D. Dissertation Dept. of Geography-Meteorology, University of Kansas, Kansas City.
- Opheim, H. (1982), "Fast Data Reduction of a Digitized Curve," Geo-Processing, Vol. 2, pp. 33-40.
- Ramer, Urs (1972). "An Iterative Procedure for the Polygonal Approximation of Plane Curves," Computer Graphics and Image Processing, Vol. 1, No. 3, pp. 244-256.
- Reumann, K. and A. Witkam (1974), "Optimizing Curve Segmentation in Computer Graphics," International Computing Symposium, Amsterdam, Holland, pp. 467-472.
- Roberge, J. (1985), "A Data Reduction Algorithm for Planer Curves," Computer Vision Graphics and Image Processing, Vol. 29, pp. 168-195.
- Robinson, A. H. and B.B. Petchenik (1976), "The Nature of Maps," University of Chicago Press, Chicago.
- Rosenfeld, A. and E. Johnston (1973), "Angle Detection on Digital Curves," IEEE Transactions on Computers, Vol. C-22, pp. 875-878.
- Sklansky, J. and V. Gongalez (1980), "Fast Polygonal Approximation of Digitized Curves," Pattern Recognition, Vol. 12, pp. 327-331.
- Solovitskiy, B.V. (1974), "Some Possibilities for Automatic Generalization of Outlines," Geodesy, Mapping and Photogrammetry, Vol. 16, No.3.
- Spath, H. (1974), Spline Algorithms for Curves and Surfaces Unitas Mathematica

- Publication, Winnapeg, Translated from German by W.D. Hoskins and H.W. Sagar. Steward, H.J. (1974), *Cartographic Generalization: Some Concepts and Explanation*, Cartographica Monograph No. 10., University of Toronto Press.
- Sukhov, V. I. (1970), "Application of Information Theory in Generalization of Map Contents," *International Yearbook of Cartography*, Vol. 10, pp. 48-62.
- Thapa, K. (1987), "Critical Points Detection: The First Step To Automatic Line Generalization." Report Number 379 The Department of Geodetic Science and Surveying, The Ohio State University, Columbus, Ohio.
- Uotila, U.A. (1986), "Adjustment Computation Notes," Dept. of Geodetic Science and Surveying, The Ohio State University.
- Wall, K. and P. Danielsson (1984), "A Fast Sequential Method for Polygonal Approximation of Digitized Curves," *Computer Vision Graphics and Image Processing*, Vol. 28, pp. 220-227.
- White, E.R. (1985), "Assessment of Line Generalization Algorithms Using Characteristic Points," *The American Cartographer* Vol. 12, No. 1.
- Williams, C.M. (1981), "Bounded Straight Line Approximation of Digitized planer Curver and Lines," *Computer Graphics and Image Processing*, Vol. 16, pp. 370-381.

TRANSPUTER BASED PARALLEL PROCESSING FOR GIS ANALYSIS: PROBLEMS AND POTENTIALITIES

Richard G. Healey and Ghazali B. Desa
Regional Research Laboratory, Scotland
Department of Geography
University of Edinburgh
Drummond Street
Edinburgh EH8 9XP
Scotland, U.K.

ABSTRACT

The availability of parallel processing computers based on large number of individual processing elements, offers the possibility of multiple orders of magnitude improvement in performance over the sequential processors currently used for GIS analysis. Before this potential can be realized, however, a number of problems must be addressed. These include assessment of the relative merits of different parallel architectures, choice of parallel programming languages and re-design of algorithms to allow effective distribution of the computational and i/o load between individual processors, so performance can be optimized. These problems are examined with particular reference to transputer-based parallel computers and some possible GIS application areas are discussed.

INTRODUCTION

The limitations of serial processors for handling computationally intensive problems in fields such as fluid dynamics, meteorological modelling and computational physics are well-known, but it is only comparatively recently that attention has been turned to this problem in the fields of remote sensing/image processing and GIS (Yalamanchi and Aggarwal 1985, Dangermond and Morehouse 1987). Parallel processing techniques, where one or many computational tasks are distributed across a number of processing elements, have been proposed as a solution to the problem (Verts and Thomson 1988). They offer the potential for orders of magnitude improvement in performance, which should allow real-time processing of very large datasets, with powerful modelling and visualization capabilities.

Since parallel processing hardware is still at an early stage of development and parallel programming methods are distinctly in their infancy, it is difficult to make firm statements about how GIS might avail itself of this new technology. More appropriate at this stage is an examination of several aspects of the overall problem. These aspects include evaluation of existing types of hardware and software for parallel processing and

approaches to the re-design of GIS algorithms, so they can take advantage of these novel machine architectures. This paper addresses these issues with particular reference to parallel processing based on networks of transputers.

TYPES OF PARALLEL ARCHITECTURE

It is not the intention here to give an extended survey of parallel architectures, as several of these are already available (Bowler et al. 1987a, Treleaven 1988), so a brief outline of the major types will suffice to provide the context for the present discussion.

SIMD and MIMD parallelism

One major approach to the design of a parallel computer is to link processing elements (PEs) into a two-dimensional array. If a SIMD (single instruction, multiple data-stream) method of operation is used, each program instruction is despatched simultaneously to each PE which executes it on the data it has stored locally. Examples of such machines include the NASA Massively Parallel Processor, the Connection Machine and the AMT distributed array processor. MIMD (multiple instruction, multiple data-stream) machines have grown in importance of recent years because of the availability of cheap but powerful microprocessor chips. Programs running on these machines are executed by all the PEs, but at any moment each processor may be at a different stage of program execution.

Shared and distributed memory systems

MIMD machines can be sub-divided on the basis of how memory is allocated to individual microprocessors. If a number of these are connected to a number of memory modules by means of a switch, to form a common global memory, the machine is of the shared memory type. An example of this is the BBN Butterfly which utilizes Motorola 68000 or 68020/68881 chips. In a distributed memory system each PE comprises a microprocessor with its own local memory. Hardware switches or links connect the individual PEs. An example would be the CALTECH Mark III Hypercube, which is also based on up to 128 Motorola 68020/68881 chips with additional I/O processors and Weitek floating point units.

Fixed and reconfigurable architectures

A further sub-division of distributed memory MIMD machines can usefully be made, depending on whether the topology of the links between the individual PEs is fixed in the hardware or is electronically re-configurable. The latter introduces a much greater degree of flexibility into the ways the computing resource can be utilized for different applications. Examples of fixed and reconfigurable architectures are the Intel iPSC-VX, based on 80286/80287 processors, and the Meiko Computing Surface based on transputers.

TRANSPUTER-BASED PARALLEL PROCESSING

Since it is less widely known than the Motorola or Intel chip sets, it is useful to outline some of the particular features of the Transputer, which was designed with parallel processing in mind, before assessing some of the advantages and disadvantages of different parallel architectures.

The most recent version of the INMOS transputer, the T800 chip, contains a number of processing components. The first of these is a 10-MIPS 32-bit RISC processor linked at 80 MByte/Sec to 4K of on-chip RAM. In addition there is an integral 64-bit floating point unit capable of 1.5 Mflops. The chip has 4 20 MBit/sec INMOS links which can be directly connected to other transputers, together with an external memory interface with a bandwidth of 26.6 MByte/sec. The T800 is claimed to achieve more than five times the performance of the Motorola 68020/68881 combination on the Whetstone benchmark (Bowler et al. 1988).

The major vendor of transputer-based parallel computers is the UK firm Meiko Ltd. which has developed a modular, extensible and reconfigurable computer system called the 'Computing Surface'.

COMPARISON OF PARALLEL ARCHITECTURES

Given the difficulties of comparing supposedly similar serial processors, it is not surprising that comparison of parallel architectures, where there are many more parameters to consider, is a rather inexact science. Nonetheless, some major points relating to the different categories described above can be identified.

With respect to SIMD and MIMD architectures, testing of distributed array processors against transputer networks indicates that the former operate best with strongly structured algorithms which do not have independently branching chains of instructions. Requirements for very fast I/O and data comparison operations also favour SIMD machines. The two types both perform well for sorting, but transputer networks are superior for 3-d graphics and modelling requirements (Roberts et al. 1988).

For shared and distributed memory systems the picture is less clear, because shortcomings of specific hardware configurations may be compensated, to varying degrees by the use of different operating systems and programing methods. Several points need to be considered, however. Firstly, access to local memory is on average three times faster than to a global or shared memory. Secondly, the speed of interprocessor communication and thirdly, the relationship between the computational performance of each PE and the speed of interprocessor communication need to be taken into account. The usefulness of a particular architecture will then depend on the extent to which a

given problem can be decomposed into separate computational tasks, or requires access to a shared database of information (Ballie, 1988). The overall aims will be to minimize communications bottlenecks between processors, to get maximum utilization of each PE during the computation and to attain as nearly as possible a linear speed-up in processing time, as the number of processors used for a particular set of calculations is increased. Distributed memory systems, such as the transputer network, are proving popular because of fast memory access, but difficulties may arise for such systems in terms of communications overheads, if large amounts of data require to be transferred between processors.

Comparison between fixed and reconfigurable architectures is more straightforward, in that the latter is undoubtedly to be preferred, for two main reasons. Firstly, it allows the machine to 'mimic' other architectures, such as a SIMD or a hypercube, for comparative purposes. Secondly, it permits the overall computational resources to be divided into 'domains' of different sizes, so parts of the machine can be used for development while others are engaged in large scale computation.

Finally, the cost-effectiveness of parallel architectures involving large numbers of PEs, compared to single or multiple vector processors much be addressed. The parallel approach has an initial advantage because it is scalable from a small number of PEs, costing a few thousand dollars, to top-end Computing Surfaces costing several millions. In addition, using the unit comparison of megaflops/megadollar, a transputer-based machines seems to have approximately a five-fold improvement in cost effectiveness compared to a CRAY X-MP/48, with similar floating point performance (Bowler et al. 1987b).

These considerations indicate that reconfigurable, distributed memory MIMD machines such as the Meiko Computing Surface have a wide range of advantages, with potential limitations only in relation to interprocessor communication and extreme high-end requirements for floating point performance. As a result, the Meiko machine was chosen as the basis for the Edinburgh Concurrent Supercomputer Project. The first phase of this has resulted in the installation of a Meiko machine with 200 transputers, each with 4 MB of local memory, together with a filestore and specialized graphics peripherals. Future project phases will allow the installation of large numbers of additional transputers, until the machine reaches its target size of at least 1024 processors, with 10,000 MIPS total processing power, 4 GBytes of distributed memory and an expected rating in excess of 1 Gflop. This will make it one of the most powerful supercomputers in Europe. The machine is jointly managed by the University Computing Service and the Department of Physics, but is in use by a variety of other research groups also.

PARALLEL PROGRAMMING LANGUAGES

Although parallel hardware has demonstrably reached the stage where a range of applications for large scale GIS processing can be envisaged, the position is less clear in terms of operating systems and programming languages.

Since parallel machines generate many new problems of system management, they have tended to be built with special purpose operating systems, which is a first major obstacle to software development activity of any kind! In the case of the Meiko this problem has now been resolved by the development of a UNIX System V compatible operating system for development work.

The second obstacle is the lack of support for parallel programming constructs in existing languages used for GIS software, specifically FORTRAN, C and to some extent PASCAL, although FORTRAN 8X is expected to include such facilities in the future. At present, there are three alternative methods of circumventing the problem:

- i) Addition of new constructs into language compilers running on parallel machines
- ii) Use of new languages, such as ADA or OCCAM, which support parallel programming directly
- iii) Use of existing languages within a communications 'harness' provided by languages like OCCAM to allow access to parallel facilities.

In relation to the first alternative, a 'parallel' C compiler has been announced for the transputer, but problems of standardization are likely to plague this approach. The second alternative offers promise because the transputer was designed to run OCCAM, a language for parallel programming based heavily on Hoare's work on communicating sequential processes (Jesshope 1988). The language has particular strengths in facilities for message passing along OCCAM channels, but is limited in its support for the variety of data structures found in existing sequential languages. ADA, by contrast, supports parallel programming through its tasking model, while having a wealth of data structures. It also provides constructs to support the use of sound software engineering techniques (Sommerville and Morrison 1987). An ADA compiler for the transputer is currently under development and this avenue for future work will be explored when the tools become available. The final alternative is the one which is most heavily used at present on the Meiko processor, particularly for FORTRAN in an OCCAM harness. The FORTRAN implementation allows block input and output of data arrays between FORTRAN programs running on different processors, across OCCAM channels communicating over the transputer hardware links. This approach is satisfactory in the short term for testing alternative parallel algorithms or re-using existing code, to take advantage of the substantial

increase in processing power on a parallel machine. It is not, however, a suitable approach for the implementation of large programming projects, designed to produce reliable and maintainable software that makes effective use of parallel processing techniques.

The shortcomings in available facilities and standards for parallel programming languages require to be addressed urgently if progress in the use of the hardware is not to be hampered. While these shortcomings make it difficult to port existing packages, the situation may not be entirely disadvantageous, as it allows attention to be focussed at present on research into methods of parallelization of algorithms. Development of effective approaches for different kinds of algorithms is fundamental to the proper utilization of parallel processing techniques.

PARALLELIZATION OF ALGORITHMS

It is apparent from the earlier discussion that algorithms appropriate for one kind of parallel architecture may be unsuitable for another. This examination of parallelization methods will be restricted to distributed memory MIMD machines.

Although architecture dependent at the level of specific implementation, several general points about the relationships between serial and parallel algorithms can still be made (Miklosko and Kotov 1984):

- i) Effective serial algorithms may not contain any parallel elements
- ii) Some apparently serial algorithms may contain significant hidden parallelism
- iii) Non-effective serial algorithms can lead to effective parallel algorithms

Past experience and well-tried serial methods of programming may not therefore be a good guide to parallel algorithm design and the qualities of inventiveness and imagination may be of more value in developing new approaches to problem solving! There are, nonetheless, several broad approaches to algorithm parallelization, including

- i) Event parallelism
- ii) Geometric parallelism
- iii) Algorithmic parallelism

Event parallelism

This is one of the most straightforward ways of exploiting parallel processing. The approach utilizes the concept of a master processor which distributes tasks to slave processors and assembles the computational results from each in turn. Such a configuration is usually termed a

'task farm' (Bowler et al. 1987b). In the simplest kinds of problem, each slave processor runs the same code against its own specific dataset. When the processor array is large and each processor individually very powerful, as on the Meiko machine, it can be difficult to achieve data input rates sufficiently high to keep the machine busy. Conversely, where the computational load is very high but the data input more restricted the Meiko delivers extremely high performance. A good example of this type of problem is ray tracing for the display of complex 3-d objects. Since each light ray being followed is independent, the algorithm is highly parallel. While such algorithms have important application for visualization problems in GIS, many other GIS processing requirements are not of this form.

Geometric parallelism

This is a very natural kind of parallelism for GIS algorithms, as it requires that the problem space be divisible into sub-regions, within which local operations are performed. Calculations performed on elements near the boundaries of sub-regions will generally require information from neighbouring sub-regions. This emphasis on algorithm localization matches an approach which can be found, for instance, in the Intergraph TIGRIS system, for interactive editing of topological data structures (Herring 1987) and in the inward spiral algorithm for TIN generation (McKenna 1987).

Since individual sub-regions will be handled by different processors, boundary data must be passed between them, introducing a communications overhead. It is therefore important, with the current level of development of distributed memory MIMD machines, to define sub-regions such that the amount of within region processing is maximized and the between region communication is minimized. It is also advisable to locate neighbouring regions on physically adjacent processors. (Bowler et al. 1988).

The four available hardware links on each transputer are sufficient for two-dimensional geometric parallelism, but even for the simplest three-dimensional case with sub-regions forming cubes, each sub-region will have six boundary faces. This can be matched in the hardware by connecting multiple transputers as 'supernodes' to yield six or more links (Jesshope 1988).

For geometric parallelism it is also necessary to consider the way in which the overall processing operation and communication between processors is organized. If a tightly synchronous model is used, a master processor communicates with each sub-region processor to determine when all have completed their local computations for a given step. At this point an exchange of boundary information takes place before proceeding with the next computational step. This approach generally produces

inefficient hardware utilization. The loosely synchronous model, where boundary information is exchanged as soon as one processor is ready to provide it and its neighbour is ready to receive it, is to be preferred if processor workloads are broadly similar. If workloads vary significantly, as might be expected for GIS applications, complex asynchronous behaviour may result, leading to unpredictable levels of inefficiency (Norman 1988). One solution to this is to recast the problem to allow improved dynamic balancing of the workload, perhaps by assigning non-contiguous portions of the overall space to individual processors. This approach has been used effectively for three-dimensional medical imaging (Stroud and Wilson 1987) and has immediate application for voxel-based processing of geosciences data (cf. Kavouras and Masry 1987).

Algorithmic parallelism

With this approach, each individual processor performs a specific task on blocks of data which pass through the processor network in a production line fashion. Though attractive as a concept, algorithmic parallelism encounters difficulties at the implementation stage which tend to reduce its efficiency. These include communication and configuration problems. In the former case, each processor has to receive different initialization instructions for its specific portion of the computational task. In the latter case, one configuration of inter-processor linkages may be appropriate for some stages of the work and inappropriate for others. This will remain a problem until the technology for dynamic reconfiguration of processor arrays becomes available. While quantitative comparisons are very hard to find, experiments at Southampton University on Monte Carlo simulations suggest that geometric parallelism gives a much better approximation to linear speed-up as the number of processors is increased, than does algorithmic parallelism (Bowler et al. 1987a). Whether a similar conclusion would apply to parallel implementations of topological map processing must be left as a subject for future investigation!

CONCLUSIONS

Parallel processing hardware is now reaching the stage where extremely high performance computers can be built in a modular and cost effective way, particularly if powerful processing chips with good communications links, such as the transputer, are used. As usual the pace of algorithm research and software development is lagging significantly behind the hardware. Of the recognised approaches to algorithm construction, geometric parallelism, possibly combined with limited algorithmic parallelism, offers the most promising route for initial work in parallel GIS processing. Early areas of investigation using the Meiko Computing Surface include computationally intensive three dimensional GIS problems and in future, polygon overlay algorithms. Beyond these a whole range of novel and indeed exciting possibilities present themselves for parallel

search on large spatial databases, real-time data structure conversion, and new approaches to visualization and user interfacing.

REFERENCES

Baillie, C.F. 1988, Comparing Shared and Distributed Memory Computers: Parallel Computing, Vol. 8, pp. 101-110.

Bowler, K.C., Kenway, R.D., Pawley, G.S. and Roweth, D. 1987a, An Introduction to OCCAM 2 Programming, Chartwell-Bratt, Lund.

Bowler, K.C., Bruce, A.D., Kenway, R.D., Pawley, G.S. and Wallace, D.J. 1987b, Exploiting Highly Concurrent Computers for Physics : Physics Today, October Edition.

Bowler, K.C., Kenway, R.D., Pawley, G.S. and Roweth D. 1988, OCCAM 2 Programming Course Notes, Publications, Dept. of Physics, University of Edinburgh.

Dangermond, J. and Morehouse, S. 1987, Trends in Hardware for Geographic Information Systems : Proceedings Auto-Carto 8, pp. 380-385, ASPRS/ACSM, Falls Church, Va.

Herring, J. 1988, TIGRIS : Topologically Integrated Geographic Information System : Proceedings Auto-Carto 8, pp. 282-291, ASPRS/ACSM, Falls Church, Va.

Jesshope, C. 1988, Transputers and Switches as Objects in OCCAM : Parallel Computing, Vol. 8, pp. 19-30.

Kavouras, M. and Masry S. 1987, An Information System for Geosciences : Design Considerations : Proceedings Auto-Carto 8, pp. 336-345, ASPRS/ACSM, Falls Church, Va.

McKenna, D. 1987, The Inward Spiral Method: An Improved TIN Generation Technique and Data Structure for Land Planning Applications : Proceedings Auto-Carto 8, pp. 670-679, ASPRS/ACSM, Falls Church, Va.

Miklosko, J. and Kotov, V.WE. (Eds.) 1984, Algorithms, Software and Hardware of Parallel Computers, Springer-Verlag, Berlin.

Norman, M. 1988, Asynchronous Communication Course Notes, Dept. of Physics, University of Edinburgh.

Roberts, J.B.G., Harp, J.G., Merryfield, B.C., Palmer, K.J., Simpson P., Ward, J.S. and Webber H.C. 1988, Evaluating Parallel Processors for Real-time Applications : Parallel Computing, Vol. 8, pp. 245-254.

Sommerville, I. and Morrison, R. 1987, Software Development with ADA, Addison-Wesley, Reading, Mass.

Stroud, N. and Wilson, G. (Eds.) 1987, Edinburgh Concurrent Supercomputer Project Newsletter No. 3, Edinburgh

University Computing Service.

Treleaven, P.C. 1988, Parallel Architecture Overview : Parallel Computing, Vol. 8, pp. 59-70.

Verts. W.T. and Lee, C. 1988. Parallel Architectures for Geographic Information Systems, Technical Papers ACSM-ASPRS Annual Convention, Vol. 5, pp. 101-107, ACSM/ASPRS, Falls Church, Va.

Yalamanchi S. and Aggarwal, J.K. 1985, Analysis of a Model for Parallel Image Processing : Pattern Recognition, Vol. 18, pp. 1-16.

UNIFORM GRIDS: A TECHNIQUE FOR INTERSECTION DETECTION ON SERIAL AND PARALLEL MACHINES

Wm. Randolph Franklin
Chandrasekhar Narayanaswami
Mohan Kankanhall
David Sun
Meng-Chu Zhou
Peter YF Wu

Electrical, Computer, and Systems Engineering Dept.,
6026 J.E.C.,
Rensselaer Polytechnic Institute,
Troy, NY, 12180, USA,
(518) 276-6077,
Internet: Franklin@CS.RPI.EDU, Bitnet: WRFRANKL@RPITSMTS,
Fax: (518) 276-6003, Telex: 6716050 RPI TROU.

ABSTRACT

Data structures which accurately determine spatial and topological relationships in large databases are crucial to future developments in automated cartography. The uniform grid technique presented here offers an efficient solution for intersection detection, which is the key issue in many problems including map overlay. Databases from cartography, VLSI, and graphics with up to 1 million edges are used. 1,819,064 edges were processed to find 6,941,110 intersections in 178 seconds on a Sun 4/280 workstation. This data structure is also ideally suited for implementation on a parallel machine. When executing on a 16 processor Sequent Balance 21000, total times averaged ten times faster than when using only one processor. Finding all 81,373 intersections in a 62,045 edge database took only 28 seconds elapsed time. These techniques also appear applicable to massively parallel SIMD (Single Instruction, Multiple Data Stream) computers. We have also used these techniques to implement a prototype map overlay system and performed preliminary tests on overlaying 2 copies of US state boundaries, with 3660 edges in total. Finding all the intersections, given the edges in memory, took only 1.73 seconds on a Sun 4/280. We estimate that the complete overlay would take under 20 seconds.

INTRODUCTION

Algorithms specific to polyline intersection are particularly important for cartographic purposes. The classic problem of map overlay is a good example where edge intersection forms the core of the algorithm. The results of this paper are also useful in diverse disciplines such as graphics and VLSI design.

We are given thousands or millions of small edges, very few of which intersect, and must determine the pairs of them that do intersect. Clearly, a quadratic algorithm comparing all $\binom{N}{2}$ pairs is not acceptable.

Useful line intersection algorithms often use sweep line techniques, such as in Nievergelt and Preparata (1982), and Preparata and Shamos (1985). Chazelle and Edelsbrunner (1988) have an algorithm that finds all K intersections of N edges in time $T = \theta(K+N\log N)$. This method is optimal in the worst case, and is so fast that it

cannot even sort the output intersections. However, this method has some limitations. First, it cannot find all the red-blue intersections in a set of red and blue edges without finding (or already knowing) all the red-red and blue-blue intersections. Second, it is inherently sequential.

Alternative data structures, based on hierarchical methods such as quadtrees, have also been used extensively, Samet (1984). They are intuitively reasonable data structures to use since they subdivide to spend more time on the complicated regions of the scene. An informal criticism of their overuse in Geographic Information Systems is given in Waugh (1986). A good general reference on cartographic data structures is Peucker (1975).

Since cartographers deal with vast amounts of data, the speed and efficiency of the algorithms are of utmost importance. With the advent of parallel and supercomputers, efficient parallel algorithms which are simple enough to implement, are gaining importance. Since this field is relatively new, few implementable algorithms exist. Some of the related parallel algorithms in computational geometry are as follows. Akl (1985) describes some parallel convex hull algorithms. Evans and Mai (1985) and Stojmenovic and Evans (1987) present parallel algorithms for convex hulls; however they require a MIMD machine, and have tested on only a few processors. Aggarwal et al (1985) give parallel algorithms for several problems, such as convex hulls and Voronoi diagrams. They assume a CREW PRAM (concurrent read exclusive write, parallel random access machine). This is a MIMD model. No mention is made of implementation. Although it is not mentioned in those papers, randomized algorithms, such as described by Clarkson (1988a), and Clarkson and Shor (1988b) appear to lend themselves to parallelization sometimes. Yap (1987) considers general questions of parallelism and computational geometry. Hu and Foley (1985), Reif and Sen (1988), and Kaplan and Greenberg (1979) consider hidden surface removal. Scan conversion is considered by Fiume, Fournier, and Rudolph (1983). For realistic image synthesis see Dippe and Swensen (1984).

This paper concentrates on an alternative data structure, *the uniform grid*. Here, a flat, non-hierarchical grid is superimposed on the data. The grid adapts to the data since the number of grid cells, or resolution, is a function of some statistic of the input data, such as average edge length. Each edge is entered into a list for each cell that it passes through. Then, in each cell, the edges in that cell are tested against each other for intersection. The grid is completely regular and is not finer in the denser regions of the data.

The uniform grid (in our use) was first presented in Franklin (1978) and was later expanded by Franklin, Akman, and Wu (1980), (1981), (1982), (1983), (1984), (1985), (1987), and Wu(1988) . The latter two papers used extended precision rational numbers and Prolog to implement map overlay. Geometric entities and relationships are represented in Prolog facts and algorithms are encoded in Prolog rules to perform data processing. Multiple precision rational arithmetic is used to calculate geometric intersections exactly and therefore properly identify all special cases of tangent conditions for proper handling. Thus topological consistency is guaranteed and complete stability in the computation of overlay is achieved.

In these papers the uniform grid was called an *adaptive* grid. However, there is another, independent and unrelated, use of the term *adaptive grid* in numerical analysis in the iterative solution of partial differential equations. Our papers present an expected linear time object space hidden surface algorithm that processed 10,000 random spheres packed ten deep in 383 seconds on a Prime 500. The idea was extended to a fast haloed line algorithm that was tested on 11,000 edges. The concept was applied to other problems such as point containment in polygon testing. Finally it was used, in Prolog and with multiple precision rational numbers in the map overlay problem in cartography.

This present paper presents experimental evidence that the uniform grid is an efficient means of finding intersections between edges in real world data. The uniform grid is similar to a quadtree in the same sense that a relational database schema is similar to a

hierarchical schema. The power of relational databases, derived from their simplicity and regularity, is also becoming apparent.

The uniform grid data structure is also ideally suited to execution on a parallel machine because of the simple data structures. Also, it is more numerically robust than sweep-line algorithms that have problems. This is of importance in the cartographic domain because numerical instability can easily introduce topological inconsistencies which tend to be difficult to rectify.

The uniform grid technique is fairly general and can be used on a variety of geometric problems such as computing Voronoi diagrams, convex hull determination, Boolean combinations of polygons, etc.

INTERSECTION ALGORITHM

Assume that we have N edges of length L independently and identically distributed (i.i.d.) in a 1×1 screen. We place a $G \times G$ grid over the screen. Thus each grid cell is of size $\frac{1}{G} \times \frac{1}{G}$. The grid cells partition the screen without any overlaps or omissions. The intersection algorithm proceeds as follows.

1. For each edge, determine which cells it passes through and write ordered pairs (*cell number, edge number*).
2. Sort the list of ordered pairs by the cell number and collect the numbers of all the edges that pass through each cell.
3. For each cell, compare all the edges in it, pair by pair, to test for intersections. If the edges are *a priori* known to be either *vertical* or *horizontal*, the vertical edges are compared with the horizontal edges only. To determine if a pair of edges intersect, we test each edge's endpoints against the equation of the other edge. We ignore calculated intersections that fall outside the current cell. This handles the case of some pair of edges occurring together in more than one cell.



Fig 1(a). USA Map - Shifted and Overlaid on itself

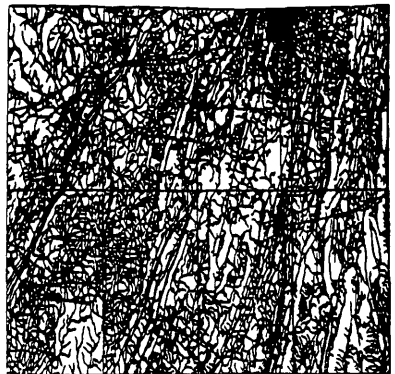


Fig 1(b). Chickamauga Area - All 4 overlays

THEORETICAL ANALYSIS

Let $N_{c/e}$ be the number of cells that an average edge passes through. The determination of $N_{c/e}$ is similar to the Buffon's Needle Problem, McCord (1964). A simple analysis shows that,

$$N_{c/e} = \left(1 + \frac{4}{\pi}LG\right) \quad (Eq.1)$$

Then N_p , the total number of (cell, edge) pairs is

$$N_p = N\left(1 + \frac{4}{\pi}LG\right) \quad (Eq.2)$$

The average number of edges per cell is

$$N_{e/c} = \frac{N_p}{G^2} \quad (Eq.3)$$

$$= \frac{N}{G^2} \left(1 + \frac{4}{\pi}LG\right) \quad (Eq.4)$$

The time to calculate the (cell, edge) pairs is

$$T_1 = \alpha N_p \quad (Eq.5)$$

where α is a constant. The time to test the edges for intersections is about

$$T_2 = \beta G^2 N_{e/c} (N_{c/e} - 1) \quad (Eq.6)$$

where β is a constant. The overhead for processing the cells is

$$T_3 = \gamma G^2 \quad (Eq.7)$$

where γ is a constant. and the total time is

$$T = T_1 + T_2 + T_3 \\ = N \left[(\alpha - \beta) + \frac{4}{\pi}(\alpha - \beta)LG \right] + \beta N^2 \left[\frac{1}{G^2} + \frac{8}{\pi} \frac{L}{G} + \frac{16}{\pi^2} L^2 \right] + \gamma G^2 \quad (Eq.8)$$

This is minimized if the 2 fastest terms in the sum grow at the same speed, which occurs when $G = \min \left[\delta \sqrt{N}, \frac{\pi}{4L} \right]$ for some δ .

What about some cells being denser since the edges are randomly distributed? Since the time to process a cell depends on the square of the number of edges in that cell, an uneven distribution might increase the total time. However, since the edges are assumed independent, the number of edges per cell is Poisson distributed, and the expected value of the square of the number of edges equals the square of the expected number of edges. Therefore the expected time doesn't increase.

RESULTS

Edge Intersection

For each data set we tried many values of G to learn the variation of time with G . Table 1 shows the results from intersecting the 116896 edges in all the 4 overlays of the Chikamauga DLG (Figure 1). There are 144,666 intersections in all, and the best time is 37 seconds with a 325×325 grid. The time is within 50% of this for grids from 175×175 up to 1000×1000 , which shows the extreme insensitivity of the time to the grid size. This is why real scenes with dense and sparse areas can be accommodated efficiently.

For the USA state boundaries shifted and overlaid on themselves, the execution time is within 20% of the optimum from about $G = 40$ to $G = 400$ and is within a factor of two of the optimum from about $G = 20$ to $G = 700$. Outside these limits, the execution time starts to rise quickly.

The economy of the grid structure is shown by the fact that the number of comparisons between pairs of edges needed to isolate the intersections is about twice the number of the edges when using the optimal grid resolution. This behavior was also observed in hidden surface algorithm described in earlier publications. There is not much room for

No. of edges	116896
Avg. edge length	0.00231
Standard deviation	0.0081
Xsects. by end pt. coincidence	135875
Xsects. by actual equation soln	8791
Total intersections	144666

Grid Size	Pairs	P/Cell	P/Edge	Grid Time	Sort Time	Xsect Time	Total Time
50	131462	52.585	1.125	4.33	3.67	182.04	190.04
80	140407	21.939	1.201	4.50	3.93	90.75	99.18
100	146389	14.639	1.252	4.72	4.22	67.31	76.25
125	153492	9.823	1.313	4.88	4.32	51.36	60.56
175	168341	5.497	1.440	5.43	4.82	36.22	46.46
200	175791	4.395	1.504	6.70	6.13	35.18	48.01
275	197815	2.616	1.692	8.45	7.68	31.78	47.91
325	212282	2.010	1.816	7.37	6.18	23.60	37.15
400	234372	1.465	2.005	8.37	7.15	21.82	37.33
500	263646	1.055	2.255	10.18	7.78	20.62	38.58
625	300413	0.769	2.570	11.72	8.92	20.22	40.85
800	351891	0.550	3.010	14.52	10.77	21.37	46.65
1000	410589	0.411	3.512	17.72	12.93	23.05	53.70
2000	704147	0.176	6.024	31.05	22.92	29.57	83.53

Table 1: Intersecting 116,896 Edges of the Chikamauga DLG

further improvement by a hierarchical method.

The largest cartographic database was the 116,896 edges of the Chikamauga Digital Line Graph (DLG) from the USGS sampler tape. The average edge length was 0.0022 and the standard deviation 0.0115, so the edges were quite variable. We used a 325×325 grid to find all 144,666 intersections in 37.15 seconds on a Sun 4/280. Other results are listed in Franklin, Chandrasekhar, Kankanhalli, Seshan, Akman (1988).

One of our examples consisted of 1,819,064 edges, with an average length of 0.0012, forming a complete VLSI chip design. We found all 6,941,110 intersections in 178 seconds. In this case, the program was optimized to use the orthogonality of the edges. The edges' lengths were quite variable, with the standard deviation being over 30 times the mean. This example illustrates the generality of this method and its applicability to other areas besides cartography.

Execution in Parallel

The uniform grid method is ideally suited to execution on a parallel machine since it mostly consists of two types of operations that run well in parallel: applying a function independently to each element of a set to generate a new set, and sorting. Determining which cells each edge passes through is an example of the former operation.

We implemented several versions of the algorithm on a Sequent Balance 21000 computer, which contains 16 National Semiconductor 32000 processors, Sequent (1986), Kallstrom(1988) and compared the elapsed time when up to 15 processors were used to

the time for only one processor, Kankanhalli (1988). We used the 'data partitioning' paradigm of parallel programming which involves creating multiple, identical processes and assigning a portion of the data to each process. The edges are distributed among the processors to determine the grid cells to which each edge belongs and then the cells are distributed among the processors to compute the intersections. Since the Sequent Balance 21000 is a shared memory parallel computer, shared data structures is the communication mechanism for the processors. The synchronization of the processors is achieved by using atomic locks. Basically, the concept of 'local processing' has been adopted in this algorithm to achieve parallelism.

There were several different ways of implementing the uniform grid data structure. First, we had a G^2MP array of cells, where G is the grid size, M is the maximum number of edges per cell per processor and P is the number of processors. However this implementation took up a lot of memory space though it obviated the use of locks. Then, G^2 array of linked lists was used. This also did not require locking but it was slow because of the dynamic allocation of shared global memory. Then it was implemented using a linked list of (*cell, edge*) pairs but this also was slow because of dynamic memory allocation. Finally a G^2M array implementation was made which used atomic locks. This implementation gave the best results.

The speedup ratios range from 8 to 13. Figure 2 shows the results from processing 3 overlays of the United State Geological Survey Digital Line Graph, totaling 62,045 edges. 81,373 intersections were found. The time for one processor was 273 seconds, and for 15 processors was 28 seconds, for a speedup of about 10. This is a rate of 7.9 million edges and 10.5 million intersections per hour. For other data sets, these extrapolated times would depend on those data sets' number of intersections per edge.

The speedup achieved for any parallel algorithm is dependent on the amount of inherently sequential computation in the algorithm, the hardware contention imposed by the competing processors, the overhead in creating multiple processes and the overhead in synchronization & communication among the multiple processes. We believe that the first factor is not dominant when using the uniform grid technique. The large speedups achieved show that the other three factors also do not affect the performance significantly. Finally, the speedup, as a function of the number of processors, was still rising smoothly at 15 processors. This means that we should achieve an even bigger speedup on a more parallel machine.

Map Overlay

We are implementing a complete map overlay package in C on a Sun workstation. The input and output are in a simplified form of the Harvard Odyssey cartographic database format. The preliminary version emphasizes clarity at the expense of speed by representing the process as a pipeline of several sequential processes. Each process writes its output to a temporary ASCII file for the next process to read, thus incurring repeated I/O costs. The stages are as follows.

1. *Deform*: In this stage chains are broken into edges.
2. *Intersect*: This stage finds all intersection points between edges.
3. *Connect*: This stage breaks up original chains into new chains, additional break points being made at the intersection points.
4. *Link*: This stage calculates and sorts the angles between each of the chains at each node and the x-axis.
5. *Form*: This stage recognizes all polygons formed by the new chains.

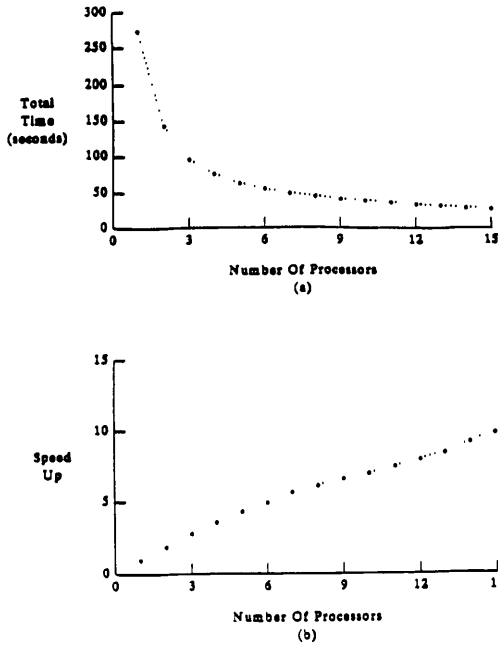


Fig 2. Time and Speedup when intersecting the 62045 edges in the Roads & Trails, Railroads and Pipes and Transmission Lines Overlays of The Chickamauga DLG in Parallel on 1 to 15 Processors. Grid size = 250. 81,373 intersections found.

6. *Display*: This displays the resulting overlaid map along with labels for each recognizable polygon.
7. *Timer*: This sums up the time each of the first 5 modules takes to complete each individual task.

One, possibly controversial, decision, was to split the chains into the individual edges at the start. This makes the data more voluminous, but much simpler, since now the elements have a fixed length. After we have intersected all the edges, and split them into pieces which are the edges of the result, it is easy to reform the output chains.

Another advantage of using individual edges is that the algorithm will be easier to

implement on a parallel machine for even greater speed.

We have implemented the algorithm partly on a Sun 3/50 and part on a Sun 4/280. Testing all 3660 edges in both input maps to find intersections takes only 1.73 seconds on the Sun 4.

CONCLUSION

Our technique has been successfully used for the important problem of map overlay which occurs in cartography. The results indicate that this is a very robust general technique which is fast and simple. It is evident from this research that simple solutions are often faster than theoretically efficient but convoluted and complicated methods. Also, the power of randomized techniques in algorithm design for real world problems is now being appreciated. Our algorithm is parallelizable and shows very good speedup with minimal auxiliary data structures.

As mentioned before, we are investigating other problems where the uniform grid technique may be applied for inventing parallel algorithms. We feel that the uniform grid technique is a good technique for parallel geometric computation in the future.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation under PYI grant no. CCR-8351942. It used the facilities of the Computer Science Department funded by a DoD equipment grant, and the Rensselaer Design Research Center.

REFERENCES

- Aggarwal, Alok, Chazelle, Bernard, Guibas, Leo, O'Dunlaing, Colm, and Yap, Chee (1985) "Parallel Computational Geometry," *Foundations of Computer Science - 25th Annual Symposium*, pp. 468-477 (1985).
- Akl, Selim G. (1985) "Optimal Parallel Algorithms for Selection, Sorting, and Computing Convex Hulls," pp. 1-22 in *Computational Geometry*, ed. Godfried T. Toussaint (1985), pp. 1-22.
- Chazelle, Bernard and Edelsbrunner, Herbert (1988) "An Optimal Algorithm for Intersecting Line Segments in the Plane," *Foundations of Computer Science - 29th Annual Symposium*, White Plains (October 1988).
- Chrisman, T.K. Peucker, and N. (1975) "Cartographic Data Structures," *The American Cartographer* 2(1), pp. 55-69 (1975).
- Clarkson, Kenneth L. (1988a) "Applications of Random Sampling in Computational Geometry, II," *Proc 4th Annual Symposium on Computational Geometry*, Urbana-Champagne, Illinois, pp. 1-11, ACM (June 6-8, 1988).
- Clarkson, Kenneth L. and Shor, Peter W. (1988b) "Algorithms for Diametrical Pairs and Convex Hulls that are Optimal, Randomized, and Incremental," *Proc 4th Annual Symposium on Computational Geometry*, Urbana-Champagne, Illinois, pp. 12-17, ACM (June 6-8, 1988).
- Dippe, Mark and Swensen, John (1984) "An Adaptive Subdivision Algorithm and Parallel Architecture for Realistic Image Synthesis," *Computer Graphics* 18(3), pp. 149-158 (July 1984).

- Evans, D.J. and Mai, Shao-wen (1985) "Two parallel algorithms for the convex hull problem in a two dimensional space," *Parallel Computing* 2, pp. 313-326 (1985).
- Fiume, Eugene, Fournier, Alan, and Rudolph, Larry (1983) "A Parallel Scan Conversion Algorithm with Anti-Aliasing for a General-Purpose Ultracomputer," *Computer Graphics* 17(3), pp. 141-150 (July 1983).
- Franklin, W. Randolph (1978) *Combinatorics of Hidden Surface Algorithms*, Center for Research in Computing Technology, Harvard University (June 1978). Ph.D. thesis
- Franklin, Wm. Randolph (1980) "A Linear Time Exact Hidden Surface Algorithm," *ACM Computer Graphics* 14(3), pp. 117-123, Proceedings of SIGGRAPH'80 (July 1980).
- Franklin, Wm. Randolph (1981) "An Exact Hidden Sphere Algorithm That Operates In Linear Time," *Computer Graphics and Image Processing* 15(4), pp. 364-379 (April 1981).
- Franklin, Wm. Randolph (1982) "Efficient Polyhedron Intersection and Union," *Proc. Graphics Interface'82*, Toronto, pp. 73-80 (19-21 May 1982).
- Franklin, Wm. Randolph (1983) "A Simplified Map Overlay Algorithm," *Harvard Computer Graphics Conference*, Cambridge, MA (31 July - 4 August 1983). sponsored by the Lab for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.
- Franklin, Wm. Randolph (1984) "Adaptive Grids for Geometric Operations," *Cartographica* 21(2 & 3) (1984).
- Franklin, Wm. Randolph and Akman, Varol (1985) *A Simple and Efficient Haloed Line Algorithm for Hidden Line Elimination*, Univ. of Utrecht, CS Dept., Utrecht (October 1985). report number RUU-CS-85-28
- Franklin, Wm. Randolph and Wu, Peter YF (1987) "A Polygon Overlay System in Prolog," *Autocarto 8: Proceedings of the Eighth International Symposium on Computer-Assisted Cartography*, Baltimore, pp. 97-106 (March 29- April 3, 1987).
- Franklin, Wm. Randolph, Chandrasekhar, Narayanaswami, Kankanhalli, Mohan, Seshan, Manoj, and Akman, Varol (1988) "Efficiency of Uniform Grids for Intersection Detection on Serial and Parallel Machines," *Computer Graphics International*, Geneva (May 1988).
- Hu, Mei-Cheng and Foley, James D. (1985) "Parallel Processing Approaches to Hidden Surface Removal in Image Space," *Computers & Graphics* 9(3), pp. 303-317 (1985).
- Kallstrom, Marta and Thakkar, Shreekanth (1988) "Programming Three Parallel Computers," *IEEE Software* 5(1), pp. 11-22 (January 1988).
- Kankanhalli, Mohan "The Uniform Grid Technique for Fast Line Intersection on Parallel Machines," (1988) M.S. Thesis, Electrical, Computer & Systems Engineering Dept., Rensselaer Polytechnic Institute, Troy, NY (April 1988).
- Kaplan, Michael and Greenberg, Donald P. (1979) "Parallel Processing Techniques For Hidden Surface Removal," *Computer Graphics* 13, pp. 300-309 (1979).
- McCord, J. and Moroney, R (1964) pp. 4-8 in *Probability Theory*, The Macmillan Company, New York (1964), pp. 4-8.

Nievergelt, J. and Preparata, F.P. (1982) "Plane-Sweep Algorithms for Intersecting Geometric Figures," *Comm. ACM* 25(10), pp. 739-747 (October 1982).

Preparata, Franco P. and Shamos, Michael Ian (1985) *Computational Geometry An Introduction*, Springer-Verlag (1985).

Reif, John H. and Sen, Sandeep (1988) "An Efficient Output-sensitive Hidden-Surface Removal Algorithm and its Parallelization," *Proc. Fourth Annual Symposium on Computational Geometry*, pp. 193-200 (June 1988).

Samet, H. (1984) "The Quadtree and Related Hierarchical Data Structures," *ACM Computing Surveys* 16(2), pp. 187-260 (June 1984).

1986. Sequent Computer Systems Inc., *Balance Technical Summary*, 1986.

Stojmenovic, Ivan and Evans, David J. (1987) "Comments on two parallel algorithms for the planar convex hull problem," *Parallel Computing* 5, pp. 373-375 (1987).

Waugh, T.C. (1986) "A Response to Recent Papers and Articles on the Use of Quadtrees for Geographic Information Systems," *Proceedings of the Second International Symposium on Geographic Information Systems*, Seattle, Wash. USA, pp. 33-37 (5-10 July 1986).

Wu, Peter Y.F. and Franklin, Wm. Randolph (1988) "A Logic Programming Approach to Cartographic Map Overlay," *International Computer Science Conference*, Hong Kong (December 1988).

Yap, Chee-Keng (1987) "What can be Parallelized in Computational Geometry?," *International Workshop on Parallel Algorithms and Architectures* (May 1987).

**A geographic data model based on HBDS concepts: The IGN
Cartographic Data Base Model**

François Salgé - Marie Noëlle Sclafer

Institut Géographique National - France
B.P. 68 - 2, Av. Pasteur
94160 Saint-Mandé
FRANCE

ABSTRACT

The cartographic data base (BDCarto) is a major IGN-France priority. Its main aim is to deliver structured geographic information with a 1:100,000 level of abstraction. It will be completed by the end of 1992 and different output products will be available for users: files, customized maps and map series (1:100,000 - 1:500,000).

The content of the data base has already been defined and the BD Carto glossary is composed of some 700 terms.

In order to define the relational form of the data base the content definition went through a structuration process. The first step was to define a geographic data model using HBDS concepts.

Roughly this model lies on two levels : the geometric level gives the geometric topology and the coordinates whilst the semantic level gives all the information about the geographic objects and their semantic topology (when necessary).

The paper describes the geographic data model and its application to the BD Carto content. It explains how the structuration process allows us to model the ground truth and get rid of the constraints of a graphical model.

THEORETICAL INTRODUCTION

GRAPH THEORY

A graph $G=(X,U)$ is the pair constituted by a set $X=\{x_1, x_2, \dots, x_n\}$ of vertices and a family $U=(u_1, u_2, \dots, u_n)$ of elements of the cartesian product $X \times X$ called arcs.

A subgraph of G is a graph having all of its vertices and arcs in G . A subgraph of G is said to be a connected component of G if for each pair of

vertices of the subgraph there exists a path joining them together using only arcs of the subgraph.

A graph is said to be planar if it is possible to represent it on a plane such that the vertices are distinct points, the arcs are simple curves and two arcs do not meet beyond their extremities.

A face of a planar graph is by definition a plane region bounded by arcs such that two arbitrary points in that region can always be linked by a continuous line in the plane which does not encounter vertices or arcs.

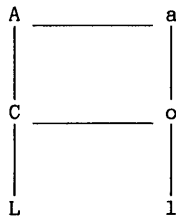
A layer is a planar subgraph of G , generated by $A \subseteq X$ such that $\forall a \in A, \forall b \in X-A$, there is no path in U linking a and b . In other word, a layer is a connected component of G or is the union of connected components of G .

HBDS CONCEPTS

HBDS stands for Hypergraph Based Data Structure. There are six basic concepts. The object concept 'o' makes it possible to define the basic entities of the considered domain. Those objects may be grouped by their characteristics; this leads to the concept of class 'C'. Those characteristics are known as attributes 'a', this defines the class-attribute 'A'. Links 'L' between classes are possible whose occurrences are known as object-links 'l'.

One can also define an hyperclass which is a set of classes. Figure 1 is a graphical summary of HBDS concepts.

Fig 1 Graphical representation of HBDS _____



MODEL OF THE BDCarto GEOGRAPHIC DATA

The terrain elements modelled in the database are described by two information levels: a geometric level which specifies their positions and a descriptive level which specifies their characteristics and their non-metric description. The entire set of data, making it possible to describe each element of the terrain modelled in the BDCarto, will be called

The following is a list of synonyms which have been used in the literature:

- vertex = point = node = junction = 0-simplex
- arc = line = edge = branch = 1-simplex

geographic data.

GEOMETRIC LEVEL

The geometric level is constructed by definition on n independent layers. Each layer lies on a planar graph constituted by arcs, vertices and faces.

For a given layer there are three classes: arcs-class, vertices-class and faces-class. Those three classes will also be called geometric classes.

Two topologic relations exist between the arcs-class and that of the vertices:

- "have for initial vertex"

- "have for final vertex"

Likewise two topologic relations exist between the arcs-class and that of the faces:

- "have for the left face"

- "have for the right face"

It is the geometric level which supports the coordinates specifying the positions of the elements of the database.

Each arc is represented by a broken line.

DESCRIPTIVE LEVEL

The object corresponds to the descriptive part of the terrain elements to be represented. They are grouped in "object-classes" (sets of objects) which form a partition of all the objects contained in the database.

An object-class is distinguished by a class name which indicates the nature of the objects contained in the class and by a list of the class-attributes, such that any object contained in the class takes for each class-attribute one and only one value amongst the possible values of the attribute (definition domain). The complete set of values taken by an object for each class attribute constitutes a description of the object.

Objects can be linked one to another. Objects can be linked to geometric elements. That dependence is expressed by relations between classes (relation in its mathematical sense, of which the arrival and departure domains are classes). There are two types of relations:

- Construction relations which make it possible to reconstruct the geometry of all the elements of the terrain modelled in the database. They link object classes amongst themselves or object classes and geometric classes.

- Semantic relations which make it possible to express a link between two objects not conditioning their geometric reconstruction. There are no semantic relations between object classes (of the descriptive level) and geometric classes.

One can distinguish two types of objects. 1) Elementary objects are directly in a construction relation with the geometric level. An element of the geometric level can be related with several elementary objects and reciprocally an elementary object can be related with several elements of

the geometric level. 2) Complex objects are constructed with elementary objects or less complex objects.

A theme is constructed by classes of objects covered by the description of part of the terrain reality. There will be semantic relations between classes of objects belonging to different themes.

A theme can correspond to several layers and a layer can correspond to several themes.

Certain classes can be regrouped for convenience into sets of classes, e.g. in order to use simplifying generic names. The HBDS diagram in Fig 2 describes the types of entities which make up the model of the BDCarto geographic data.

INFORMATION ON THE GEOGRAPHIC DATA

CONSTRAINTS ON THE GEOGRAPHIC DATA

This is the logical assertion concerning the data base entities whose value must be TRUE. In other words, the constraints are postulates which, as a rule, verify all the data in the base. There are constraints called integrity constraints in the database, which can only be true and must therefore be verified by the acquisition sub system (without exceptions).

The other constraints, semantic constraints, which can have exceptions, must preferably be evaluated before insertion.

Integrity constraints

uniqueness: any object, any element of a geometric class, any construction relation occurrence, only exists once (for a given date) in the database.

definition domain: any attribute of any object of any class exists, is unique and belongs to its attribute-class definition domain.

positioning: any object of the database is connected to the geometric level through a set of construction relations in order to produce an unique geometric representation of the terrain element which it is modelling.

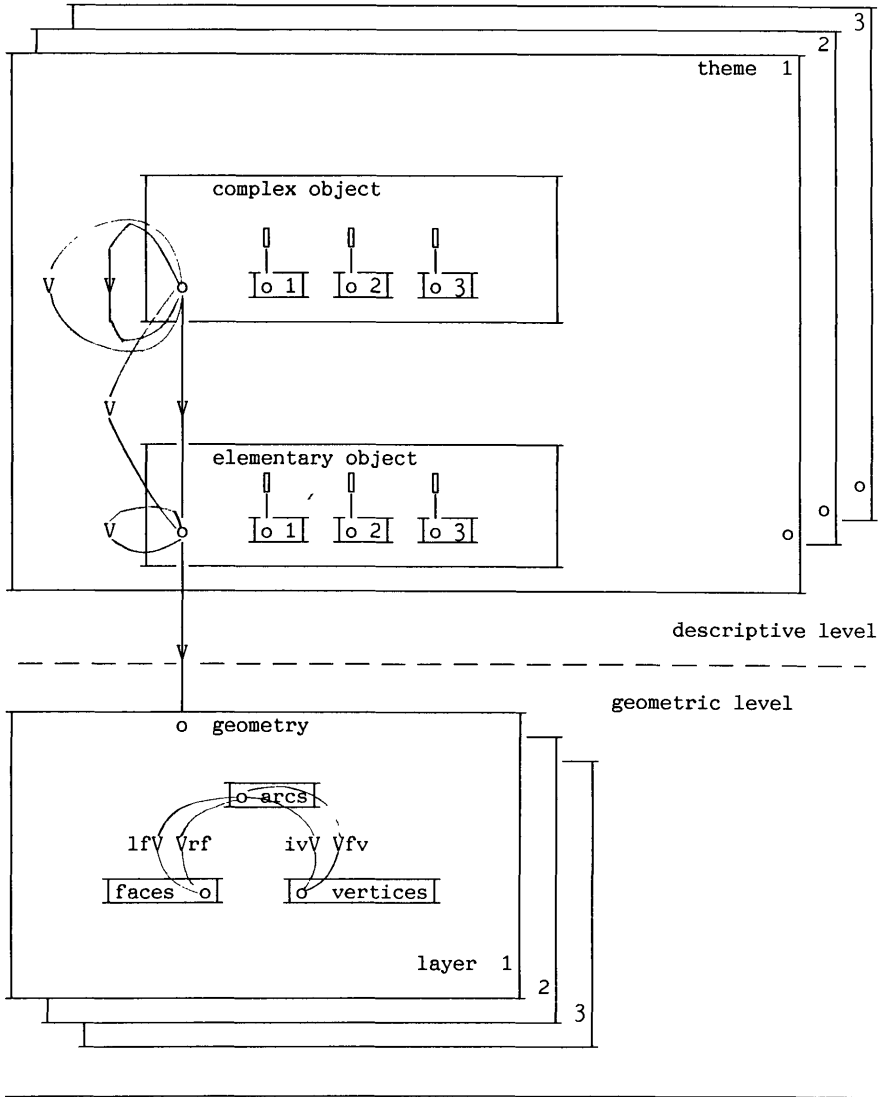
layer: the set of elements of a geometric class in the same layer constitutes a planar graph.

Semantic constraints

These constraints vary for each entity and must be compulsorily specified for each one.

object-class: constraints concerning possible combinations of attribute values for an object of the class (subset of the cartesian product of the definition domains of all the attributes of the class of objects)

Fig 2



relation between classes: constraints concerning restrictions on the departure domains and arrival domains and its properties

geometric class: constraints concerning the minimum size of arcs, faces ...

geometric construction of elementary objects: constraints on the type of geometric description of elementary objects

other constraints: constraints concerning a combination of classes and/or relations.

GENEALOGY (ORIGIN) OF GEOGRAPHIC DATA

The genealogy specifies for each geographic datum:

- the source used for the data capture, i.e.
 - type of source
 - scale of source
 - identification of the source (number or title, date of publication)
 - date of collection of the information in the field
- the process which leads to the integration of the digital data into the database, including the coordinate transformations. This process is described by reference to the specifications.

The genealogy is an information which concerns the classes of objects (from where and how does one know that the objects which constitute it really exist), the relation between classes of objects (from where and how does one know that such an object is related with such other object), the geometric construction relations (from where and how does one know that such an object can have such a geometric description), the attributes (from where and how does one know that such an object takes such a value for that attribute).

It is attached to each geographic datum.

QUALITY

Position accuracy

It is defined by two parameters: the mean error and the standard deviation between the coordinates read in the database and those of the real field point that they are supposed to represent.

That information concerns each geometric construction relation or each relation occurrence.

Semantic information accuracy

One is interested here in the difference between the values taken by the semantic information in the data base and their reality in the field. That measurement is given in the form of a probability [P(condition)]. That information concerns:

- the object-classes: accuracy of the identification of the object of the class

$P(\exists o | o \in C_o \text{ and } o \notin \bar{C}_o)$
probability that an object exists which belongs to class C_o in the

database and whose real class is not Co.

-the relations between classes: identification accuracy of the relation occurrence expressed by:

$$P(\exists o, o' \mid \neg(\overline{oRo'}) \text{ and } oRo')$$

probability that two objects exist which are related in the database and are not related in the reality.

-the attributes: accuracy of the value taken by each object for that attribute expressed by

$$P(\exists o \mid o.A \neq \overline{o.A})$$

probability that an object exist whose database attribute is o.A and its real attribute is not o.A.

Logical consistency

It verifies the respect of the semantic constraints. It is attached to each constraint C expressed in the form of a probability:

$$P(\text{objects, relations, attributes exist} \mid C = \text{FALSE})$$

It can be forced to 0.

Completeness

This concerns the exhaustivity of the information effectively acquired with respect to the database content specifications and the field reality. That measurement is given in the form of a probability. That information concerns:

-the object-classes: existence of all the objects of the class expressed by

$$P(\exists o \mid o \notin Co \text{ and } o \in \overline{Co})$$

probability that a real object exists which is forgotten in the database.

-the relations between classes: existence of all the occurrences of a relation expressed by:

$$P(\exists o, o' \mid \neg(oRo') \text{ and } \overline{oRo'})$$

probability that an occurrence of a relation has been forgotten.

-the attributes: number of the unknown value taken by that attribute expressed by

$$P(\exists o \mid o.A = \text{unknown})$$

probability that an object exists for which the value of the attribute A is not known.

Quality assessment

A deductive estimation based on a knowledge of the errors at each stage of the process, on the calibration of the geometric sources and on the hypotheses concerning error propagation can be deduced from the genealogy information and can be attached to the construction relations and to the occurrences of those relations.

The evaluation of the quality, due to its statistical nature, only has a sense with respect to a given set of information. Consequently, a quality estimate is undertaken by means of a quality report specifying:

the zone concerned by the evaluation

the content elements having been the subject of the evaluation

the date of the measurement

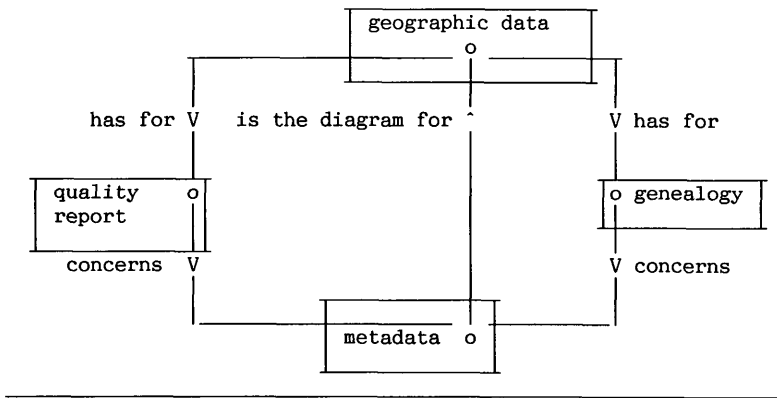
the description of the method of measurement
 the description of the external reference
 the reference validity date
 the value of the result
 the objects to which one can allocate the result.

The geographic data are structured in conformity with the model previously discussed. The description of that data structure, with which one associates the content and quality specifications, constitutes the meta-data.

SUMMARY HBDS DIAGRAM

Two sets of data are distinguished: the geographic data themselves and the information on the data. The later breaks down into three groups: the metadata and constraints, the genealogical information, the quality reports. The HBDS diagram which rules those data sets is as follows (Fig 3).

Fig 3 HBDS summary



Ref. :1 The IGN cartographic data base : From the users'needs to the relational structure - F. Salgé - Marie Nöelle Schlafer.
 in Euro Carto 7 Environmental Applications of DIGITAL Mapping - 1988

2 A survey on the HBDS methodology applied to cartography and land planning - F. Bouillé
 in Euro Carto 6 - 1987

3 The proposed standard for digital cartographic data
 in the American Cartographer Volume 15.No.1 - January 1988

**DEMONSTRATION OF IDEAS IN FULLY AUTOMATIC LINE
MATCHING OF OVERLAPPING MAP DATA**

**Raymond J. Hintz
Mike Z. Zhao**

**Department of Surveying Engineering
120 Boardman Hall
University of Maine
Orono, Maine 04469
(207) 581-2189**

ABSTRACT

The analytical matching of similar lines on a common map edge is a difficult process which has been addressed by several authors. The procedure is more complex when the information overlaps. Two examples fall in the latter situation. The first case is when different maps of a common area are being digitized into a GIS environment. A second example is digitized data in the overlap region between adjacent stereomodels in a strip of photos, or in the sidelap region between adjacent strips.

In many cases the stereoplotter operator visually controls the line matching procedure using on-line computer graphics, which contributes to an already tedious procedure. Final map "clean-up" often uses a tablet digitizer which communicates to the computer graphics system. This paper will detail prototype algorithms, which has been developed and tested in a PC environment, which can be resolve various matching problems without operator intervention. The matching procedure follows user-defined tolerance limits in its analysis, and provides error information in situations which cannot be resolved. Many of the same algorithms will be shown also as effective tools in "smoothing" the intersections of dissimilar lines.

INTRODUCTION

While it is a simple, though tedious, procedure for a human operator to match the digitized features in the overlapping region of two photogrammetric models or two adjacent map sheets, there are indeed some particular difficulties in implementing this procedure by a computer. A person has an innate ability to recognize geometric patterns and the trends which should be utilized in the matching process. While a computer is a proven tool in automatically solving large and complex numeric problems, identifying geometric structures and resolving them create a very difficult problem to computer software.

Several authors have worked on various aspects of edge matching problems [Shmutter et al., 1981], [Schenk,

1986], [Beard et al., 1988]. Discussions on the problems of matching topographic feature lines, adjusting elevations of model grids, and establishing boundaries between models were presented by Shmutter, et al. (1981). The difficulty of implementing edge matching with multiple features in a stereomodel overlap region has been discussed, and a generalized theoretical approach has been presented by Schenk (1986). Another approach for edge matching without overlap has been demonstrated by Beard, et al. (1988). The approach utilized in this study was designed for overlapping regions in addition to common edges.

The focus of this investigation is similar to the one studied by Schenk (1986). Many line features cross map or stereomodel boundaries: contour lines, roads, rivers, railways, etc. In this study it is assumed that a feature is assigned a line type and a CAM (Computer-Aided Mapping) attribute. The complex multi-feature matching problem needs to be broken into manageable small pieces so that the whole problem becomes a series of individual feature matching problems. A strongly structured programming style has been employed in solving the problem [Frank, 1987]. The goal of the project was to create a PC-environment line matching program which complements existing Kern analytical stereoplotter and mapping software. Turbo Pascal 4.0 was used as the programming language since it caters to highly modularized programming. The programs developed have been tested using both fictitious data and actual digitized data. In all situations, the developed algorithms can be easily adapted to any generalized edge matching problem.

THE NATURE OF THE MULTI-FEATURE MATCHING PROBLEM

The selected priority of line connections is outlined in Table 1.

Connection (Connector, Connectee) Priority:

1. Curved line to curved line, ends.
2. Curved line to straight line, side.
3. Straight line to straight line, ends.
4. Straight line to straight line, side.
5. Curved line to curved line, side.
6. Straight line to straight line, side.
7. Curved line to symbol (point).
8. Straight line to symbol (point).
9. Straight line to curved line.
10. Curved line to straight line, ends.

Table 1. The specifications the multi-feature matching problem.

At first glance, the requirements presented in Table 1 appear as only manipulations of the geometric relations of lines and points. It is actually the generalization of the aspects of geometry in the multi-feature matching problem. The effective classification and organization of the spatial relations of the data sets are the key to effective

management of a large amount of digital information.

The specifications in Table 1 can be generally divided into two distinct categories. Certain situation occurs only in the overlapping region of two data sets. This situation is restricted to the same line types in both data sets as in the first and third rows in Table 1. The second situation is generalized line matching of any map information. This paper addresses the later situation as a stretch/peelback ("clean" line connection) problem, and it should not be mixed with the first line matching problem. The two problems are thus addressed separately.

While a line type is straight, curved, or a symbol, the CAM mode describes the line by color, thickness, and nature (dotted, dashed, etc). Essentially only one line matching algorithm has been developed as straight lines and symbols are treated as special types of curved lines.

GENERAL SCHEME OF THE PROGRAM STRUCTURES

Two programs have been developed to solve the identified problems of line matching in an overlapping region (MODTIE) and the stretch/peelback problem (PRETIE). PRETIE can be implemented before or after implementing the edge matching program MODTIE. Since both programs handle the same geometric entities, many of the same modularized routines can be used in each. Both programs use the same format for an input connection file. An example is listed in Table 2.

CONNECTOR	CONNECTEE	TOLERANCE	CONDITION
L2, C1	L2, C1	5	C
L2, C1	L2, C1	5	E
L2, C1	L2, C1	5	S
L5, C2	L5, C2	3	M
L1, C1	L1, C1	2	M

Table 2. Example of a connection file.

In Table 2 L# is the line type number and C# is the CAM mode number. This constitutes a unique identifier for an individual line type. The programs decide which connection needs to be implemented based on the connection condition. In these conditions, S means end to side, C is self-closed, and E is an end to end connection. These conditions are resolved in PRETIE. M indicates edge matching, and is thus resolved in MODTIE.

The mechanism of program MODTIE will be described in detail in the following sections. Since program PRETIE has many similarities, a discussion of it will be generalized.

While program PRETIE operates on data in a single input file, program MODTIE operates on two input data files and creates one output file. MODTIE needs to manipulate the data only within the overlapping region of the two data sets. In photogrammetric model joins this region is relatively small since at the usual 60% overlap between

photos there is only a 10% overlap between the adjacent models. The data outside the overlap region does not need to be analyzed. The overlapping region, called a user-box, can be defined by the software automatically, or by user definition. Figure 1 represents the data separation idea, where Map1 and Map2 are the input data files, and Map3 is the output data file.

After the data in the overlapping region have been separated from the original data files, they are stored in temporary files. The software then utilizes individual line and CAM types for single feature matching in the fashion illustrated in Figure 2.

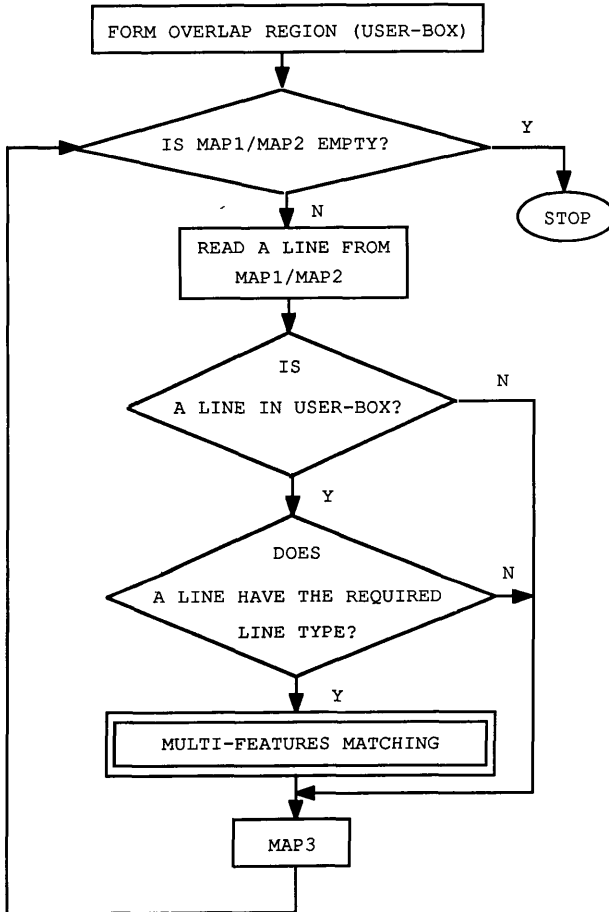


Figure 1. Data separation and analysis.

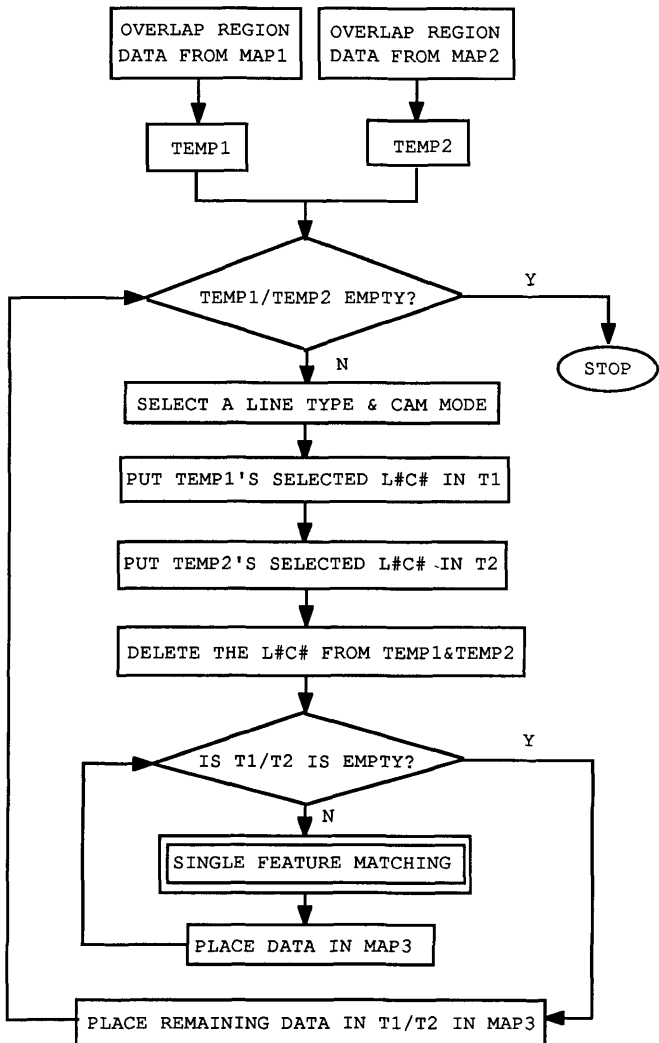


Figure 2. Break down the multi-feature problem into single feature problems.

The general structure of program PRETIE is illustrated in Figure 3.

IMPLEMENTATIONS

Identification of Conjugate Lines

The unique identifier L#C# (line type and CAM mode) allows software to concentrate on the same feature from two groups. Ideally, each line in one file, should have a corresponding line in the other file. If it is not true, there is a "dead end" to the line in the overlap region. Either this is a mistake or an actual occurrence. This situation is alerted to the operator.

To locate a pair of lines that appears suitable for connection, a search for an intersection of the two lines is conducted. Non-intersecting lines are connected if the minimum distance between them does not exceed the user-specified tolerance. A rectangle search region about the lines is created by the software (Figure 4). In addition to the distance checking, information such as elevation associated with contour lines is used to ensure a high percentage of correct identifications.

This method is successful if a user defines reasonable tolerance limits on line matches. These tolerances must be based on the quality of the data.

Based on the identification of conjugate lines, the connection between the lines is then conducted. A line can have multiple connections in the overlapping region (Figure 5). This situation further complicates the connection algorithm. To ensure a high percentage of successful connections, a general algorithm has been developed which efficiently continues to look for additional connections to the conjugate line.

A final problem which had to be resolved in connecting overlapping lines is which portion of a particular line needs to be discarded. All examples in figure 6 represent a curved line which needs to be connected, and it is obvious the connected line should pass through both the left and right ends of the overlap region. The direction of a line is arbitrary and the line connection algorithm has been developed to handle all four situations. The algorithm consists of two steps. The first step of the algorithm ensures that the two lines are merged along the original direction of the first line. The second step of the algorithm solves the ambiguity problem resulting from the four situations in Figure 6. The resolved ambiguity is represented in Figure 7 where Figure 7(a) results from 6(a) and 6(b), and 7(b) from the 6(c) and 6(d). The algorithm inverts the direction of the first line and conducts the line connection again. By comparing the length of the two connected lines, the longer one is selected as the final result.

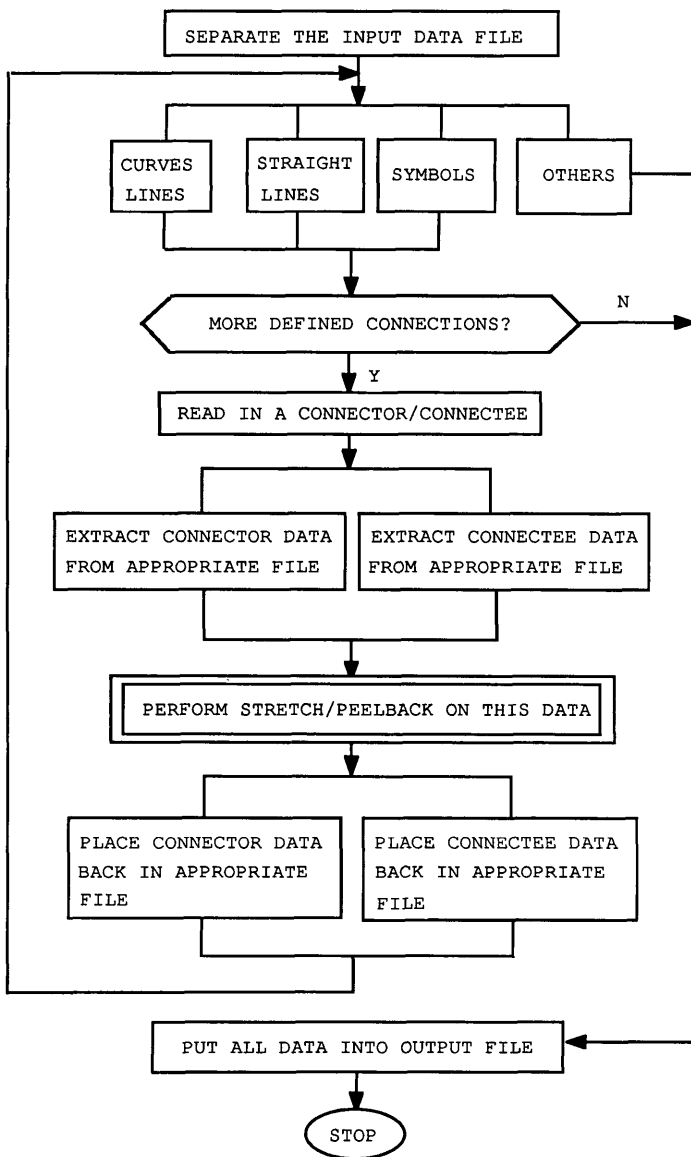


Figure 3. General flowchart of program PRETIE.

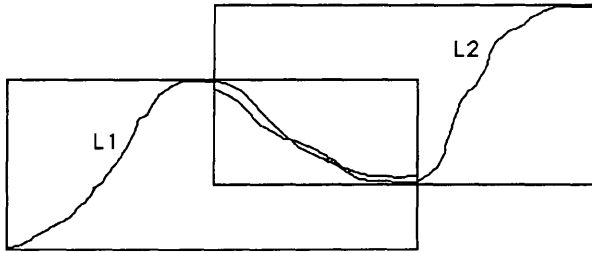


Figure 4. Rectangle search region for line match.

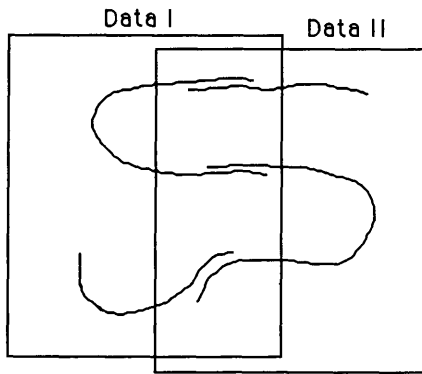


Figure 5. Multiple line matches.

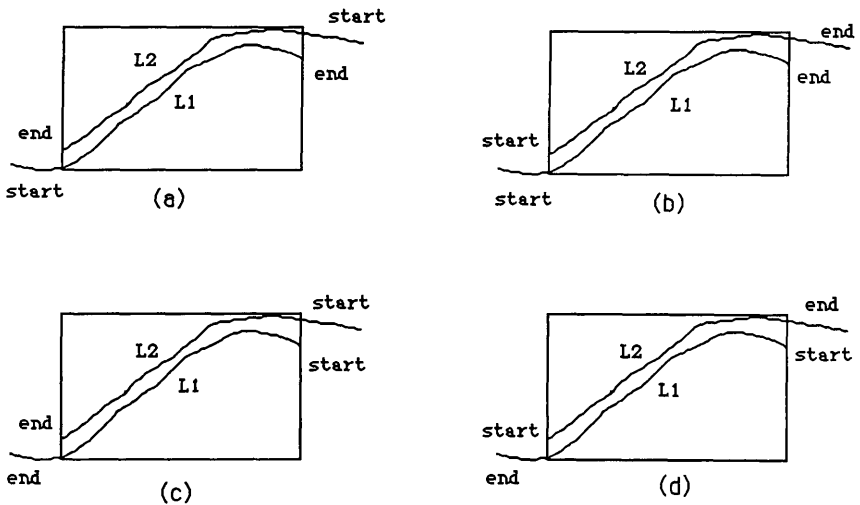


Figure 6. Four situations of the relative relationship between two conjugate lines.



Figure 7. Two possible results after connecting the lines of Figure 6.

Figure 8 demonstrates an example of matching digitized contours, where 8(a) and 8(b) are the results from two adjacent photogrammetric stereomodels, 8(c) is the combination of 8(a) and 8(b) without processing, and 8(d) is the merged results after running program MODTIE.

From Figure 8 it can be seen that there are some disconnections among the contours in the model. This is a special situation of stretch/peelback. The "smoothed" result using program PRETIE is illustrated in Figure 9.

CONCLUSIONS

A structured approach in solving the edge matching problem has been shown to be successful. The key in this approach is to break down a large and complex problem into many small ones that are manageable. With this approach a high percentage of line matching and stretch/peelback connections can be fully automated.

ACKNOWLEDGEMENTS

The authors wish to thank Kern Instruments, Inc., of Brewster, New York for their support in this research.

REFERENCES

- Beard, K.M., 1988, A Localized Approach to Edgematching, The America Cartographer, Vol. 15, No. 2, pp. 163-172.
- Frank, A.U., 1987, Geometry and Computer Graphics, Class Notes, Dept. of Surveying Engineering, University of Maine, Orono, Maine.
- Schenk, T., 1986, A Robust Solution to the line-Matching Problem in Photogrammetry and Cartography, Photogrammetric Engineering and Remote Sensing, Vol. 52, No. 11, pp. 1779-1784.
- Shmutter, B., and Y. Doytsher, 1981, Matching Between Photogrammetric Models, The Canadian Surveyor, Vol. 35, No. 2, pp. 109-119.

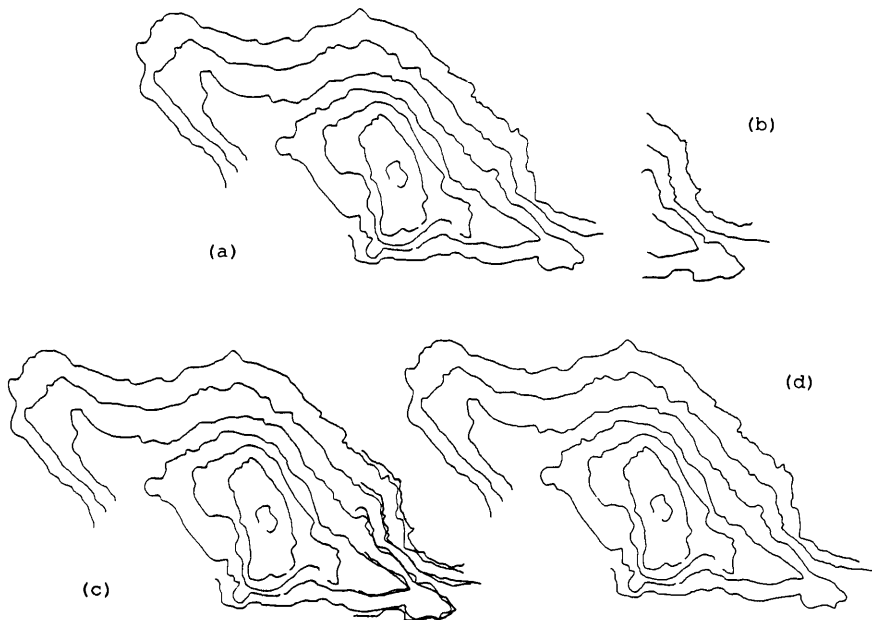


Figure 8. Example of edgematching of overlapping photogrammetric digitized data.

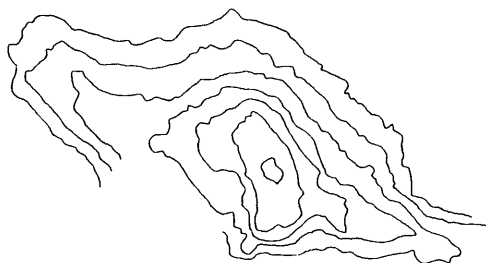


Figure 9. Example of data "smoothing" of digitized photogrammetric data.

TOPOGRAPHIC GRAIN AUTOMATED FROM DIGITAL ELEVATION MODELS

Richard J. Pike¹, and William Acevedo²
U.S. Geological Survey
Don H. Card²

National Aeronautics and Space Administration
¹Menlo Park, CA 94025; ²Ames Research Center, Moffett Field, CA 94040

ABSTRACT

Relief at topographic grain is an estimate of local relief optimized by varying unit-cell size. In homogeneous terrain, local relief (Y) within nested circles increases with circle size (X) and then levels off at a diameter termed "grain," a measure of characteristic local ridgeline-to-channel spacing. To map relief and grain as continuous variates, we automated their estimates from digital elevation models (DEMs). The computer calculates values of elevation dispersion within nested sample areas in a DEM, plots them against sample size, and analyzes this function to identify the Knick, or break-point. The resulting quantities grain and relief at grain appear to correspond to "range" and "sill", two parameters of spatial autocovariance.

INTRODUCTION

Once-intractable problems in regional geomorphology and physiography are beginning to yield to analysis of digital elevation models (DEMs) manipulated on fast computers by spatial-analysis software. A long-standing goal in landform interpretation is to abstract the character of continuous topography (Pike, 1988). Numerical methods for such representation of terrain require measures of land form that minimize chances of misinterpretation and can be readily communicated and mapped. Many parameters have been devised to describe topographic geometry, at different scales, in both horizontal (XY) and vertical (Z) domains (Evans, 1972). This is our first report on experiments with the automation of two related measures, grain and local relief.

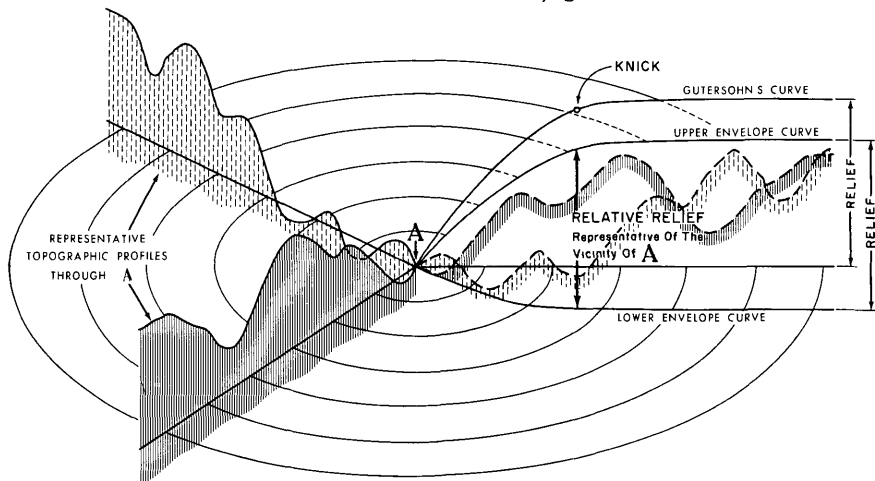


Figure 1. Gutersohn's (1932) concept of local relief measured on varying areas, here circles, optimized at a Knick resulting from the addition of two height envelopes (from Thompson, 1959, 1964).

RELIEF AND GRAIN

Topographic grain is the characteristic horizontal spacing of major ridges and valleys. Grain is inherent in Johnson's (1933) restricted definition of texture as "the average size of the units comprising a given topography." The grain concept arose from the need for a variable and nonarbitrary unit-cell size within which to calculate another parameter, local relief, rather than from any perceived need to measure texture per se (Johnson, 1933). Defined as elevation range ($Z_{\max} - Z_{\min}$) within a limited area, local relief has a serious operational drawback: estimates of it from unit cells of one size do not represent a wide spectrum of terrain types with equal fidelity (Trewartha & Smith, 1941; Wood & Snell, 1960). This problem reflects the varied dominance of topography by local features that differ widely in relief and spacing (Johnson, 1933; Thompson, 1959, 1964).

Gutersohn (1932) devised a calculation for local relief such that size of the unit cell would be neither arbitrary nor uniform. According to his concept, envelope curves of maximum and minimum elevation vary with distance in a way that defines the optimal areas for measuring local relief (Figure 1). Unused until its adoption by Wood and Snell (1959) to optimize the sample design for their oft-cited quantitative taxonomy of terrain (Wood & Snell, 1960), the method entails measuring relief in nested squares centered at a sample point on a topographic map and plotting relief (Y) against length of the side of the square (X). In homogeneous topography, relief generally increases rapidly with size of the square until the full range of local elevation has been encountered, after which it increases much more slowly. The cell size corresponding to the relief value at this breakover, or inflection, is large enough, but no larger than required, to include the most important features typifying that topography (Figure 2).

The varying area of a topographic sample, adjusted "for the degree of coarseness or fineness of the relief pattern" (Trewartha & Smith, 1941), seems to have been termed "grain" by someone at the University of Wisconsin, likely before its use by Young (1954). We think that W.F. Wood, Young's contemporary at the Department of Geography at Madison, adopted "grain" for his implementation of Gutersohn's (1932) approach to calculating local relief (Wood & Snell, 1959, 1960). Neuenschwander's (1944) review of morphometric analysis included Gutersohn's breakthrough, which evidently was first described outside the German literature by Hook (1955), a student of W.F. Wood at Iowa State University. Hook did not mention "grain." The choice of terms was unfortunate, for "grain" conflicts with accepted usage describing map patterns and trends. However, it is so well entrenched in the literature that we decline to propose any alternative here.

PAST WORK

In practice, grain varies widely with topographic texture. Values obtained by Wood & Snell (1960) from 1:100,000-scale contour maps (n=413 samples) in central Europe range from 2 to 14 miles. Thompson's (1959, 1964) study of the Alps from 1:250,000-scale maps (n= nearly 300), by a method differing from Gutersohn's in practice but not in concept, yielded grain measurements between 2 and 28 miles. A regional analysis of southern New England by the Wood-Snell method (Pike, 1963) from 1:24,000-scale USGS quadrangles (n=142) resulted in grain values of 1 to 11 miles. Grain varied from 1 to 6 miles in Georgia at 1:62,500 (n=76) and southern New York at 1:24,000 (n=94) (Autometric, 1964). All these studies used circular samples.

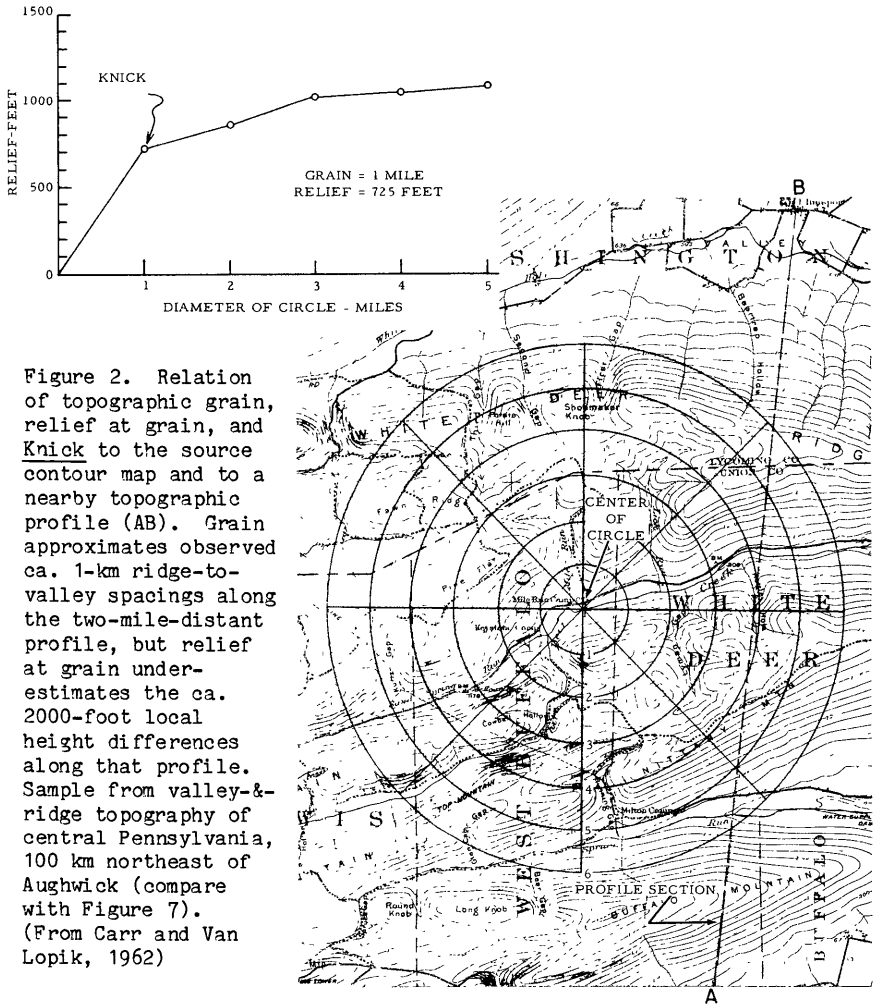
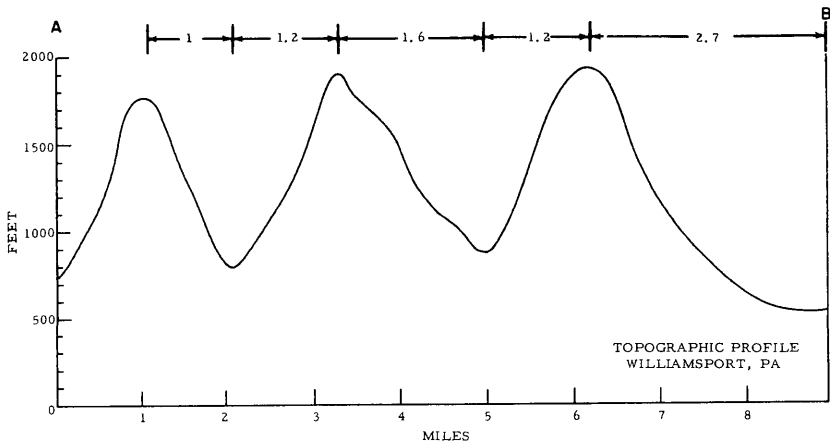


Figure 2. Relation of topographic grain, relief at grain, and Knick to the source contour map and to a nearby topographic profile (AB). Grain approximates observed ca. 1-km ridge-to-valley spacings along the two-mile-distant profile, but relief at grain underestimates the ca. 2000-foot local height differences along that profile. Sample from valley-&-ridge topography of central Pennsylvania, 100 km northeast of Aughwick (compare with Figure 7). (From Carr and Van Lopik, 1962)



Although grain is an important descriptor of meso-scale topographic texture, its relation to other attributes of land form differs so much, and so unsystematically, by locale and map scale that its geomorphic significance has never been properly ascertained. The only grain results that have been mapped and contoured are those from the Alps (Thompson, 1959, 1964) and southern New England (Pike, 1963). In the latter region grain correlates strongly with local relief and mean elevation ($r = 0.79$ and 0.67 , respectively, from unpublished results).

Grain measurement has problems, aside from the obvious tedium of the technique. Its results are unrepresentative in heterogeneous (nonstationary) topography (where a sample includes contrasting physiographic units), in very low-relief terrain, or where the sample center lies near the intersection of major valleys or ridges (Wood & Snell, 1960). Even under favorable conditions, the relief/distance curve may not inflect crisply, and its visual appraisal can be subjective. Pike (1963) found that using circle area rather than diameter often sharpened the inflection, or Knick (plural Knicks), literally a break or bend (Gutersohn, 1932), of the relief/distance function. However, this modification does not remove all ambiguity.

Grain was first automated for topographic profiles, rather than areas, by Pike 20 years ago (Schaber et al., 1980). The algorithm was part of a terrain-analysis package inspired directly by the early DEM work of Tobler (1968). The method computes relief in nested segments of a sampling traverse (beginning in its center), plots relief against segment length, draws the relief/distance curve, and selects as grain the segment length where convex change in slope along the curve is sharpest (the Knick). Automated values of grain for 12 profiles on 1:62,500-scale maps (Pike, 1988) range from 6 to 26 km.

THIS STUDY

We are developing algorithms to automate estimates of grain and relief over areas within large DEMs, thus complementing the automation of relief and other measures for invariant sample cells (Pike & Acevedo, 1988). The procedure follows that devised for profiles. The computer searches successively larger nested circles or squares around a point in a DEM, computes relief or another measure of elevation dispersion, and graphs the results (Y) against the corresponding sample sizes (X). To reduce subjectivity in selecting the Knicks, we fit the relief/sample-size curve with many pairs of complementary, linear, intersecting equations. Topographic grain is defined, on the X-axis, at the intersection of the pair of equations that minimizes least-squares. Relief at grain is defined, on the Y-axis, as the corresponding value of local relief. Maps of grain and relief at grain result from moving the sampling procedure through a DEM.

We automated the analysis of grain on two datasets. Developmental work was done on 15 new minimal-error USGS 1:24,000-scale DEMs (derived from digitized contours, rather than from stereo profiling) for San Mateo County, California (resolution 30m). Further tests of automated against manual grain values were run on 1:250,000-scale data from the Defense Mapping Agency Topographic Center (DMATC) digital terrain tapes of southern New England (63-m-resolution). We used Sun 3/260* and 4/260* workstations and our own software.

*Trade names and trademarks in this paper are for descriptive purposes only, and imply no endorsement by the U.S. Geological Survey.

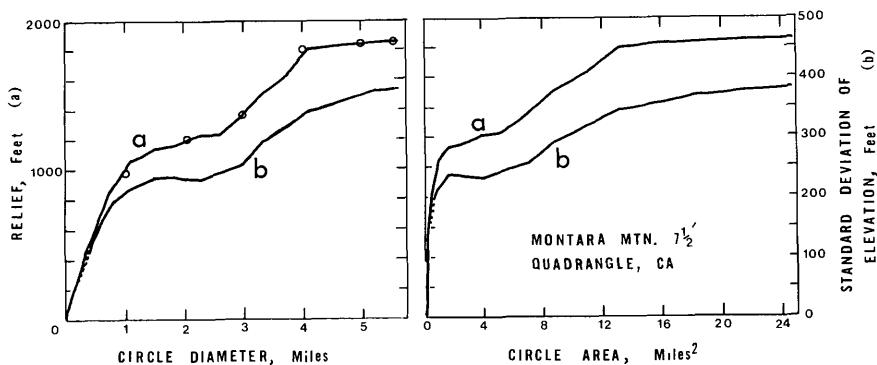


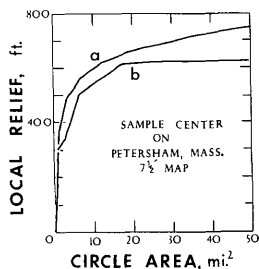
Figure 3. Automated grain curves from 30-m DEM: Relief (a) and standard deviation of elevation (b), each as a function of sample diameter (left) and area (right). Manual measurements on map (dots) yield a similar curve. Grain here is about 1 - 1.5 miles.

The experiments addressed three main issues: (1) Obtaining the curve of relief/sample size from DEMs automatically, (2) Determining how to best locate the optimum inflection, or Knick, on this curve, and (3) locating the Knick automatically, without human judgment.

RESULTS: STANDARD GRAIN ANALYSIS

The first tests showed that manual and automated techniques yield virtually identical relief/sample-size curves from 1:24,000-scale maps and DEMs (Figure 3), when sample locations and cells are similar (squares or circles). Curves differed more in tests of 1:250,000-scale DMATC data (automated) against 1:24,000-scale maps (Pike, 1963, manual) (Figure 4), which we ascribe to the contrasting information content of the data. Quite different curves resulted where sample locales differed or if one method used squares and the other circles.

Figure 4. Relief/area curves for same locale but different methods and data. (a) Manual method: 1:24,000-scale maps (Pike, 1963). (b) Automated method: DMATC 1:250,000-scale digital data. In both cases grain occurs at a circle diameter of about 3 to 4 miles and a relief at grain of about 600 feet.



The next tests suggested that such robust statistics of elevation dispersion as variance and standard deviation (Figure 3) yield at least as sharp Knicke as local relief (elevation range), for both circle diameter and area. Elevation range can be unrepresentative (because chances of including an unrepresentative height value are so high; Wood & Snell, 1960; Evans, 1972), even though it may reflect the land surface more faithfully than other parameters (just as modal elevation always indicates such observed features as terraces, flood plains, and accordant summits, whereas the mean may not).

Plotting local relief or standard deviation as a function of window area, as opposed to diameter, also sharpens the Knick in automated determinations of grain (Figures 3, 5). We tried to enhance the Knick even more, by the first derivative of the curve taken at 5-pixel windows along it (Figure 5). Lastly, we attempted an optimal solution for the Knick by least-squares fits to the relief/sample-size curve. The example in Figure 5, centered in the La Honda CA 30-m DEM, used square windows increasing in edge length by 60-m increments. Results suggest a roughly 1.0-km-diameter grain and a 220-m relief at grain.

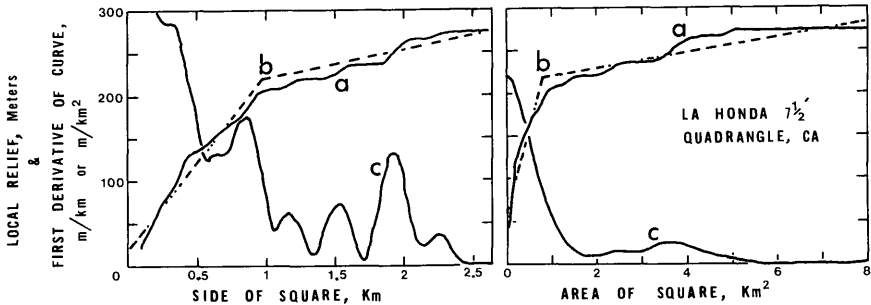


Figure 5. Topographic grain, for relief/distance (left) and relief/area (right), from three different automated calculations on 30-m DEM data: (a) relief/sample-size curves, (b) least-squares fit to curves, and (c) first derivative of curves. Grain is about 1 km and relief at grain is about 220 meters (see text).

The most robust values of topographic grain seem to result from plots of local relief (Figure 5) or standard deviation of elevation (Figure 3) against window area, plus choice of the Knick by least-squares. The area-versus-distance comparison in Figure 5 shows that least-squares analysis (b) yields the most similar grain values (0.1 km apart) among the three pairs of curves: For distance, $X = 1.0$ km; for area, X (distance equivalent of area) = 0.9 km. The curves from first derivatives (c) are the least satisfactory. That for circle diameter is too irregular to yield an unambiguous Knick and the (much smoother) curve for area yields a high grain of 1.5 km; the two grain values are at least 0.4 km apart. Knicke in just the raw relief/sample-size curves (a), at roughly 0.8 km distance and 1.1 km area, are between the two in definition (that derived from area is the sharper); the two grain values are a high 0.3 km apart. These tests suggest that the first derivative of the curve may not supply the desired enhancement.

We are not wholly satisfied that least-squares fitting yields optimal grain values. Further tests, from eight samples of DMATC data in southern New England, show that automated grain values from the least-squares technique do not always coincide with those selected by eye from the same relief/sample-size curves. Ambiguity in the computer arises from weak or multiple Knicke and from the absence of uniform criteria for graph coordinates; choices of vertical and horizontal scales critically affect the shape of the curve, in both visual and least-squares analyses. These problems have been evident since the work of Wood and Snell (1960), Thompson (1959, 1964), and Pike (1963).

Automating the procedure on DEMs enables grain and relief at grain to be mapped regionally at various resolutions. Figures 6b and c are the first maps of grain ever made by machine. We calculated 63,504 grain

values for the Montara Mountain 7.5' quadrangle in San Mateo County CA, using least-squares to identify the Knick, by moving 13 nested circular windows (diameter increment 0.65 miles) through the DEM one pixel (30 m) at a time. Grain values are low, ranging from 1.1 km (dark tones) to 2.9 km (light tones). Overall pattern reflects dominant ridgelines and stream channels as well as other contrasts in the local landforms, notably that between linedated topography to the northeast and more randomly oriented terrain to the southwest.

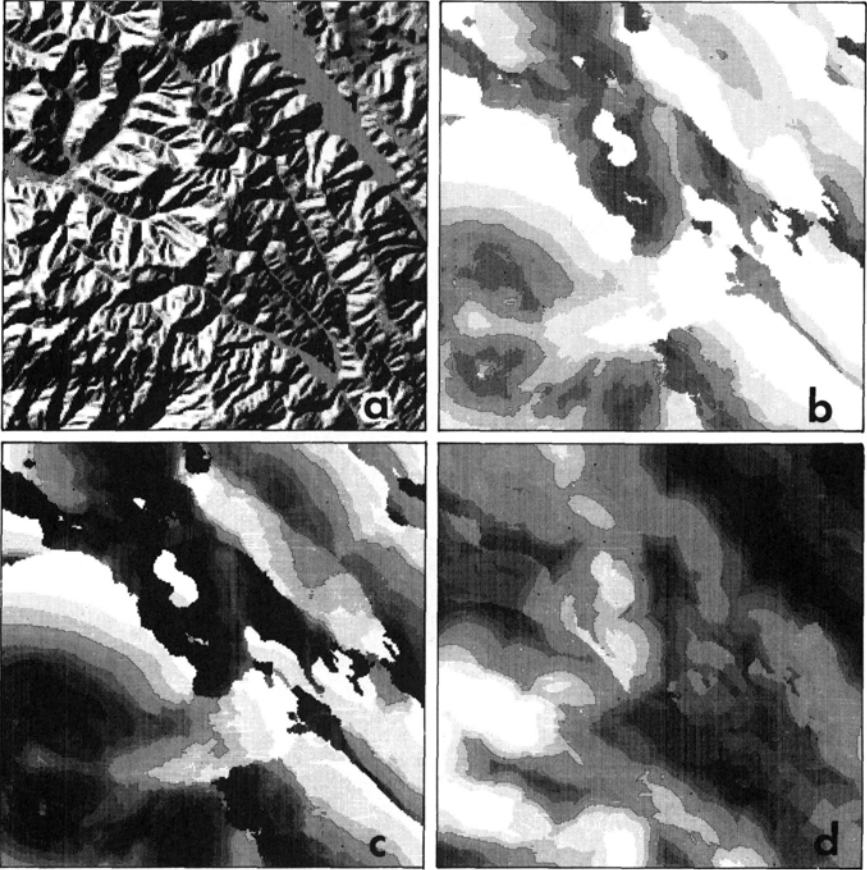


Figure 6. Maps of topographic grain (b, expressed as circle diameter; c, as circle area; see text) and relief at grain (d). Dark tones, low values; light tones, high values. Shaded relief image (a) of Montara Mountain, CA, quadrangle. The images, made from a 30-m-resolution DEM by automated methods on a Sun workstation and a Calcomp plotter, are 7.56 km across.

Figure 6d is the map of the accompanying values of relief at grain, on circles that vary from about 1 km to 3 km across. Like a slope map, Figure 6d numerically expresses the roughness of the Santa Cruz Mountains in this area. Relief at grain varies from 50 m (darkest tone) to 475 m (lightest tone). Because both this map and those of grain were made at maximum resolution (30 m), to produce fine-grained

images, the analysis is highly CPU intensive. The three maps together required 50 hours on the Sun 3/260 or 8 hours on the 4/260.

RESULTS: GRAIN AND GEOSTATISTICS

The fact that topographic elevation is a regionalized variable (Olea, 1977) leads us to believe that the methods of geostatistics (e.g., Oliver & Webster, 1986) apply directly to the problem of topographic grain. Accordingly, we are experimenting with the measurement of grain from autocorrelograms and variograms. In the first phase of this work published variograms are used to test our main hypothesis: that the relief/sample-size function yielding topographic grain is similar to the variogram of elevation for the same area. A variogram is a plot of squared differences between paired observations (Y), averaged by distance bins, against distance between those observations (X). We think the geostatistical parameters termed "range" and "sill" on elevation variograms (Olea, 1977) are equivalent to grain and relief at grain (Figure 7) and that both methods describe the same attribute of topography, spatial autocovariance of elevation.

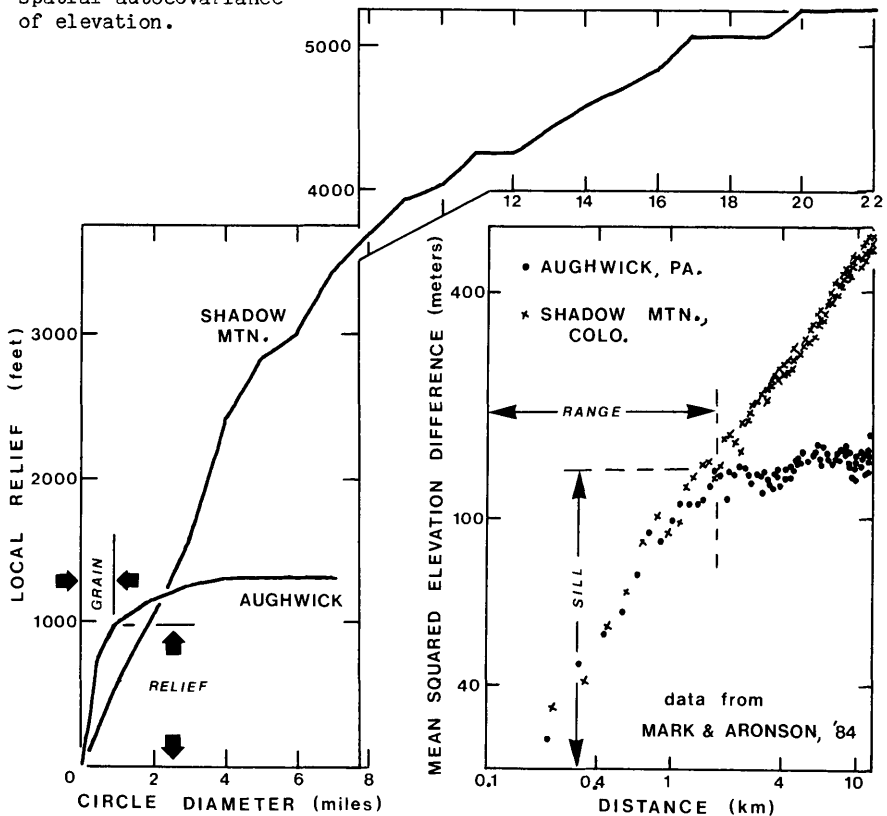


Figure 7. Correspondence of grain and relief at grain to their geostatistical equivalents "range" and "sill" near Aughwick, Pennsylvania (see Figure 2). Manual relief/diameter curves from 1:250,000-scale maps (left); automated variograms (right) from 1:24,000-scale DEMs (Mark & Aronson, 1984) for same areas. Grain of Colorado sample is not reached until about 10 miles (16 km).

Variograms computed from 1:24,000-scale DEMs (Figure 1 of Mark & Aronson, 1984) are consistent with relief/distance functions of the same areas on 1:250,000-scale contour maps (Figure 7). We manually measured grain (1-mile-circle increment) for Mark & Aronson's Aughwick Pennsylvania and Shadow Mountain Colorado samples, the only variograms showing data points. Both the Aughwick grain curve and its variogram inflect at a distance of one mile \pm 0.5 mi. The variogram for Shadow Mountain does not inflect, but that is only because its computation stopped at a distance of 12.5 km (7.8 mi.) and the Knick, which is not crisply defined in this area, does not occur until at least a 9- to 11-mile-circle diameter (Figure 7). These comparisons suggest that grain values might be obtained from topographic variograms, and perhaps more objectively than is possible by the traditional procedures.

W.F. Wood long ago contended that autocorrelation ultimately would be the best way to estimate topographic grain (personal communication, 1964). We believe that our results, however preliminary, confirm his view. Perhaps more important, formal geostatistics may supply a much-needed theoretical basis for the relief/grain concept.

CONCLUSIONS

Topographic grain, a threshold phenomenon of spatial autocorrelation, measures the areal dominance of terrain by its characteristic local relief. Grain and relief at grain can be computed automatically from DEMs and mapped regionally. Automated grain values agree with those derived manually under similar conditions. Subjectivity in selecting the Knick is reduced by plotting relief against sample area instead of diameter and by least-squares partitioning of the relief/sample-size function. Additionally, standard deviation of elevation and the raw relief/sample-size function yield crisper Knicke, respectively, than relief (elevation range) and the first derivative of the curve. Lastly, although variograms may replace relief/distance functions for estimating grain and relief at grain, much further work remains before these and other issues attending topographic grain, to say nothing of its significance for landscape evolution, are understood and solved.

ACKNOWLEDGMENTS

We thank C.M. Wentworth and S.D. Ellen for reviewing the manuscript and Matt Ma, of TGS Technology Inc., for programming assistance.

REFERENCES

- Autometric Facility, 1964, Expanded Study of the Applicability of Multifactor Computer Programs to Terrain Analysis, Final Report, ONR Task 387.030, Contract Nonr 4523(00), for Office of Naval Research by The Raytheon Co., Alexandria, Va., 78 pp.
- Carr, D.D., & Van Lopik, J.R., 1962, Terrain Quantification Phase I: Surface Geometry Measurements, U.S. Air Force Cambridge Res. Lab. Rept. 63-208 (contractor: Texas Instruments, Dallas, TX), Bedford, MA.
- Evans, I.S., 1972, General Geomorphometry, Derivatives of Altitude, and Descriptive Statistics: in R.J. Chorley (ed.), Spatial Analysis in Geomorphology, London, Methuen, pp. 17-90.
- Gutersohn, H., 1932, Relief und Flussdichte, Ph.D. dissertation, Univ. Zürich, 89 pp.

Hook, J.C., 1955, The Relationship Between Roughness of Terrain and Phenomena Related to Agriculture in Northeastern United States, unpublished Ph.D. dissertation, State Univ. of Iowa, Iowa City.

Johnson, D., 1933, Available Relief and Texture of Topography A Discussion: Jour. Geology, Vol. 41, pp. 293-305.

Mark, D.M., & Aronson, P.B., 1984, Scale-dependent Fractal Dimensions of Topographic Surfaces: An Empirical Investigation, with Applications in Geomorphology and Computer Mapping: Mathematical Geology, Vol. 16, pp. 671-683.

Neuenschwander, G., 1944, Morphometrische Begriffe, Eine Kritische Übersicht auf Grund der Literatur, Ph.D. dissertation, Univ. Zürich, Emil Rüegg publ., 135 pp.

Olea, R.A., 1977, Measuring Spatial Dependence with Semivariograms, No. 3, Series on Spatial Analysis, Kans. Geol. Survey, Lawrence, 29 p.

Oliver, M.A., 1986, Semi-variograms for Modelling the Spatial Pattern of Landform and Soil Properties: Earth Surface Processes and Landforms, Vol. 11, pp. 491-504.

Pike, R.J., 1963, Landform Regions of Southern New England, A Quantitative Delimitation, M.A. thesis, Clark Univ., Worcester, MA, 80 pp.

Pike, R.J., 1988, Toward Geometric Signatures for Geographic Information Systems: Proc. Int'l. Symposium Geogr. Info. Systems, Wash., D.C., -- The Research Agenda, NASA, in press.

Pike, R.J., & Acevedo, W., 1988, Image-processed Maps of Southern New England Topography: Geol. Soc. America Abstracts with Programs, Vol. 20, No. 1 (Northeast Section annual meeting, Portland ME), p. 62.

Schaber, G.G., Pike, R.J., & Berlin, G.L., 1980, Terrain-Analysis Procedures for Modeling Radar Backscatter: Radar Geology, An Assessment, Jet Prop. Lab. Publ. 80-61, Pasadena, Calif., pp. 168-199.

Thompson, W.F., 1959 & 1964, Determination of Spatial Relationships of Locally Dominant Topographic Features: Geol. Soc. America Bull., Vol. 70, No. 12, Pt. 2, p. 1814 (abstract); also U.S. Army Natick Labs. Technical Report, Natick, Mass., 24 pp.

Tobler, W.R., 1968, A Digital Terrain Library, Tech. Rept. 08055-1-T, U.S. Army Res. Office (Durham) Contract DA-31-124-ARO-D-456, 23 pp.

Trewartha, G.T., & Smith, G.-H., 1941, Surface Configuration of the Driftless Cuestaform Hill Land: Annals Assoc. Amer. Geogr., Vol. 30-31, pp. 24-45.

Wood, W.F., & Snell, J.B., 1959, Preliminary Investigations of a Method to Predict Line-of-Sight Capabilities, U.S. Army Quartermaster Res. & Engineer. Command Res. Study Rept. EA-10, Natick, Mass., 16 pp.

Wood, W.F., & Snell, J.B., 1960, A Quantitative System for Classifying Landforms, U.S. Army Quartermaster Research & Engineering Command Report EP-124, Natick, Massachusetts, 20 pp.

Young, R.N., 1954, A Geographic Classification of the Landforms of Puerto Rico, unpubl. Ph.D. thesis, Univ. Wisconsin, Madison, 160 pp.

A SPATIAL LOW-PASS FILTER
WORKING FOR TRIANGULAR IRREGULAR NETWORK (TIN)
AND RESTRICTED BY BREAK LINES

Zi-Tan Chen, Ph.D
Environmental Systems Research Institute
380 New York St., Redlands, CA 92373, USA
Phone: (714) 793-2853

ABSTRACT

Spatial low-pass filters are widely used for smoothing 3-D surfaces. However they only work on raster data structure. A new approach here presents a spatial low-pass filter which works directly on triangular irregular network (TIN) structure. This approach has two goals. The first goal is reducing data structure conversion. It avoids a conversion between TIN and raster for smoothing. The second goal is to restrict filtering by break lines. Comparing with conventional direction-homogenous smooth procedures, this filter provides a capability to limite filtering region.

INTRODUCTION

Spatial filtering

A spatial object can be converted from spatial domain to frequency domain, and vice versa. For an one-dimensional object $f(x)$, we have

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)$$

where a and b are Fourier factors.

$$a_n = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos nx \, dx \quad (n = 0, 1, 2, \dots)$$

$$b_n = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin nx \, dx \quad (n = 1, 2, \dots)$$

This formula shows us that a spatial object can be represented as a sum of a series items in frequency domain. Each item corresponds a component with one spatial wavelength. These items are ordered from long wavelength to short wavelength in the series. Also, this formula shows that the whole series of components in frequency domain can be reversed to the original object in spatial domain.

Theoretically, information has not been lost during a conversion in either direction, if both conversions carry complete components, i.e. all items. However, if we only convert part of items from frequency domain to spatial domain, the restoring object is also an incompleated object. Two properties should emphasis on the incomplete conversion. First, the more items are counted to convert, the more precise object

can be restored. Second, ignoring different part of components generated various effects. All kinds of filters use the phenomena.

In frequency domain, details of a spatial object are represented by short wavelength components, while large features are represented by long wavelength components. If we only convert long wavelength components from frequency domain to spatial domain, then only those features with the long wavelengths are restored in spatial domain, while all details of the spatial object are eliminated. The procedure is called low-pass filter. The effect is smoothing.

There are two purposes for user to use low-pass spatial filtering (smoothing) in applications. The first is generalization. It means that we only use a limited number of items from the beginning of the series in frequency domain to instead of all items of the whole series. The restored spatial object from the part of items is similar to the original one in overall, except those very details. The second is noise (e.g. random errors) removing. It is useful when we are aware that some noises (e.g. random errors) exist in original data. Usually noises have shorter wavelengths. They only damage the accuracy of last items in frequency domain. If we select a wavelength threshold longer than the wavelength of all noises, those noises can be deleted from the original data after a low-pass filtering.

In many applications, user needs different degree of smoothing. This option can be obtained for user by selecting a wavelength as a threshold. It corresponds that user determines how many items are counted in frequency domain. Any detail with wavelength shorter than the threshold in frequency domain is erased. Only those features with longer wavelength in frequency domain are kept and restored in spatial domain.

In this way, user can control how strongly they want to smooth the surface. If a very short wavelength is selected, little smoothing is done, result is very similar to the original surface, except very details disappeared. In an extreme case, when the wavelength is zero, the filtered result is identical to the original data. If select a wavelength threshold longer than the dimension of the research area, result is just a flat surface with average elevation.

All concepts from previous discussion are also good for two and three dimension situations, although their formulas are not so simple.

Algorithm of filters depends on data structure

There are many well-known algorithms developed for implementing spatial filtering, such as Fast Fourier Transform (FFT), Fast Walsh Transform (FWT), etc. However they only work for raster data structure. This limitation causes conversion between different data structures, because once a smooth operation is necessary, the data must be converted to raster data structure.

A spatial filter on polygons was proposed in 1986 by author. It directly works on vector data structure. More particularly, a spatial

filter presenting in this paper directly works on TIN structure.

Break lines and filtering

The feature of using break lines is a capability of local control. A spatial filter working with break lines has two properties. It can recognize existence of break lines, and it can be effected by break lines. Thus, for example, a lake surface can be kept flat while the new filter is executed around it because the boundary of the lake, as a break line, can prohibit the smoothing effect into the lake region.

A LOW-PASS SPATIAL FILTER ON TIN

Algorithm

The algorithm of the filter proposed here is similar in concept to an algorithm for raster structure. For a point (X0,Y0) on the surface of a TIN, its filtered elevation ZF (X0,Y0) is an average of elevations of its neighbor points.

$$ZF (X0,Y0) = \frac{\sum_{i=1}^N Z_i}{N} \quad (i = 1, \dots, N)$$

Where Z_i is the elevation of the i th neighbor point, N is the amount of points counting in as neighbor. These counted neighbor points should be in a nearby range, because if two points are separated far away, they have little effect on each other. This range can be a horizontal circle whose center is at the point and its radius is given by user's option. The radius is actually the threshold wavelength of the filter. The larger the radius is, the more details are erased.

$$ZF (X0,Y0) = \iint_C Z(x,y) dx dy / \pi R^2$$

where $(x - X0)^2 + (y - Y0)^2 \leq R^2$

here R is the radius of the circle C .

Differently from raster data structure, the sum of elevations of neighbor points is essentially the volume of the range, because TIN represents a continuous surface.

$$dV(x, y) = Z(x, y) dx dy$$

Thus the formula can be represented as:

$$ZF (X0,Y0) = \iint_C dV(x,y) / \pi R^2$$

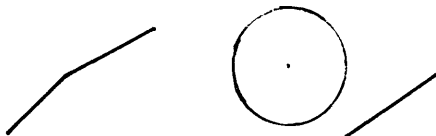
where $(x - X0)^2 + (y - Y0)^2 \leq R^2$

This formula provides a basic calculation of filtered value for any point on the TIN surface. Then a filtered surface can be produced from the input surface.

RESTRICT FILTER OPERATION BY BREAK LINES

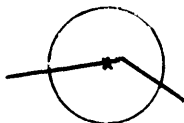
Break lines are linear feature with more precise data. A terrain surface should exactly follow break lines wherever they exist. Filtering should not effect the break lines. The break lines should have a power to restrict filtering. More disscussion about break lines is in reference (Chen, 1988).

When we use low-pass filter, for those points that none of break lines passes in its neighbor circle range, the filter result should not been influenced by any break lines. The result point value is exactly generated by the previous calculation without any consideration of break lines.

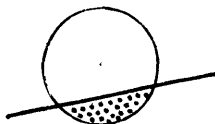


When a break line passes through its neighbor circle range, two major cases are in consideration.

First case, if the break line exactly passes the center point, the point should have an identical elevation value of the point on the break line, according to the fact that break lines are more precise. Thus, wherever the break line exists, we use break line elevations replace old values along each break line.



If a break line does not pass through the center point, but cuts a part of the neighbor circle range, as shown on the following picture, the effects from the shaded area should not be counted to influence the filtered value of the center point. In this way, the filter effects do not pass over any break line. A break line prohibits filtering from one side to another side.



EXPERIMENTS

The test area is in Misato, Japan. Original data are digitized from a contour map. Only the reservoir and the dam have more precise engineering data. They are collected as break lines.

Figure 1 shows a 3-D view of the terrain surface after a smoothing procedure. This surface is directly filtered from TIN structure. The smoothing threshold wavelength is 50 meters. We can see that the smoothing procedure influences the whole area including the reservoir. For example, the dam is too fuzzy to see.

Figure 2 shows a map of break lines at the area. These break lines are reservoir boundary and dam edges, etc.

Figure 3 shows a 3-D view at the same area, but is a result of smoothing procedure restricted by break lines. The difference of existence of the dam between two views is obvious.

CONCLUSIONS

Two goals of the proposed spatial filter are reached. A TIN surface can be filtered in TIN structure. Also, break lines can restrict the filter procedure. Further observation will be on its performance.

REFERENCES

- Chen, Zi-Tan, "Break Lines on Terrain Surface", Proceedings of GIS/LIS'88 conference, San Antonio, Texas. Dec. 1988, pp.781-790.
- Chen, Zi-Tan, "Contour Generalization by a 3-Dimensional Spatial Low-pass Filtering", Proceedings of Second Annual International Conference, Exhibits and Workshops on Geographic Information Systems, GIS'87, San Francisco, Oct. 1987, pp.357-386.
- Chen, Zi-Tan, "Spatial Filtering of Polygon Data", Proceedings of Second International Symposium on Spatial Handling, July 5-10, 1986, Seattle, pp.86-101.
- Huang, T.S., ed., 1975. Picture Processing and Digital Filtering, New York, Springer-Verlos.
- Johnson, D.E., 1976. Introduction to Filter Theory, NJ., Prentice-Hall, Englewood Cliffs.
- Tobler W.R., 1969, "Geographical filters and their inverses", Geographical analysis, 1,3, pp.236-253.

Figure I. A 3-D view of filtered terrain surface in Misato, Japan
(Using the Low-pass filter with a threshold wavelength 50 meter)

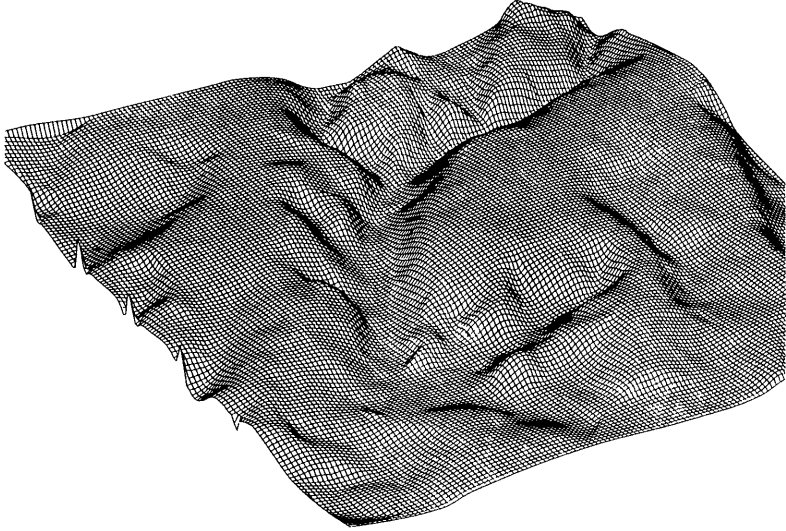


Figure II. A map of break lines in Misato reservoir area.

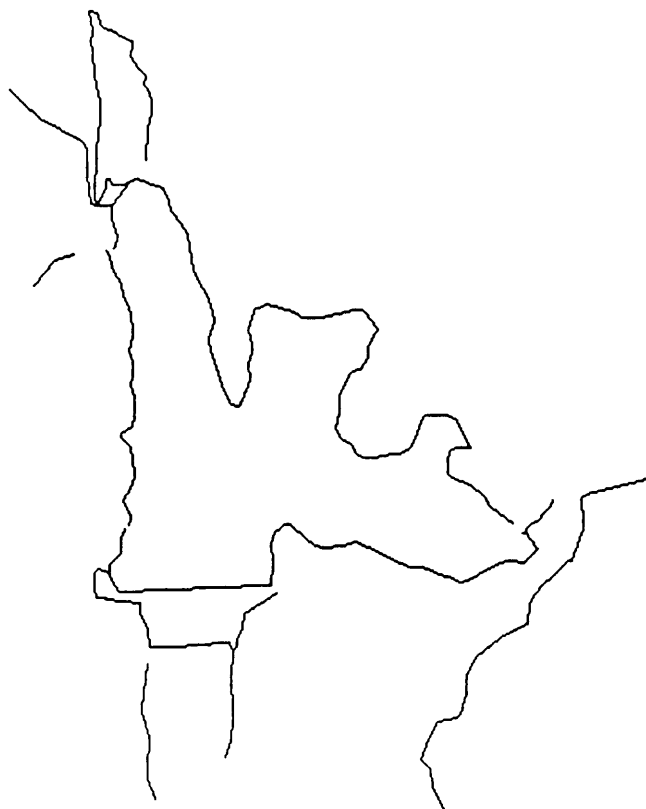
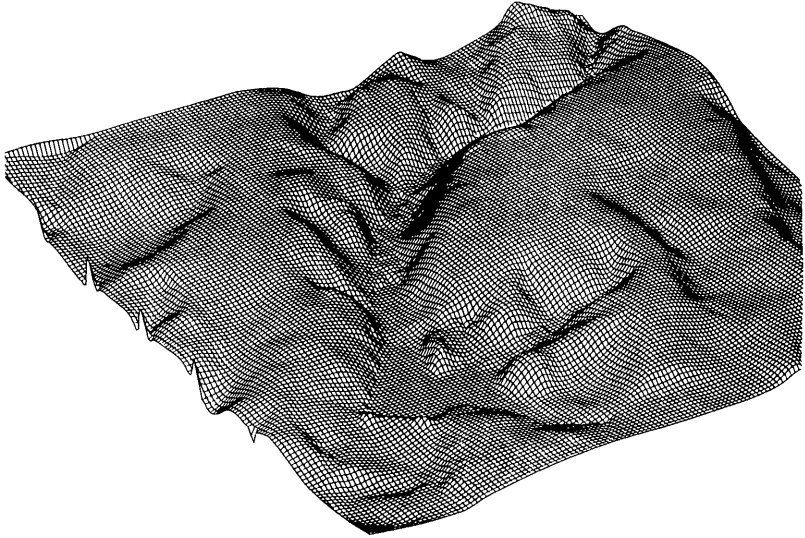


Figure III. A 3-D view of filtered terrain surface in Misato, Japan
but effected by break line on Figure II.
(Using the Low-pass filter with a threshold wavelength 50 meter)



A COMPACT TERRAIN MODEL BASED ON CRITICAL TOPOGRAPHIC FEATURES

Lori L. Scarlatos
Grumman Data Systems
1000 Woodbury Road
Woodbury, NY 11797

ABSTRACT

A broad range of applications, from military programs to survey and land use systems, rely on Digital Terrain Models (DTMs) for timely and accurate information. As more and more applications make use of this data, demands for both greater land coverage and finer, more accurate details are on the increase. Meeting these requirements can result in vast volumes of data which strain the memory limits of a computer system. Large digital elevation models can also create a bottleneck in the input/output processes and 3-D perspective rendering algorithms. Therefore, algorithms that generate compact and accurate elevation models are an important topic for research. We present one such algorithm here.

This paper describes a triangulation method which builds a DTM from a series of critical line features such as elevation contours, ridge and valley lines, and other breaklines. The method described is an improvement over current techniques because it triangulates any set of critical lines without human intervention, retains the original lines in the triangulation, adds no more than four points to the data, and runs relatively fast. Implementation results are given at the conclusion of the paper.

INTRODUCTION

Digital Terrain Models (DTMs) contain important topological information for applications such as 3-D terrain modeling, simulation, navigation, hydrology studies, visibility calculations, and route planning. For all of these applications, increasing demands for both greater land coverage and finer, more accurate details result in greater data volumes. For example, a typical DTM covers a one degree cell, which is about 3600 square nautical miles, an area smaller than Connecticut. With the elevations sampled every 3 arc seconds, or slightly less than 100 meters, this DTM occupies about 3 megabytes of memory. Elevations sampled at 10 meter intervals, a more desirable resolution for applications that rely on the DTM's accuracy, will occupy 300 megabytes for the same coverage. The United States covers over 3 million square nautical miles, so a data base for that area will grow correspondingly. This increase in data volume can strain the memory limits of a computer system. Large volumes of data also create a bottleneck in the input/output processes and 3-D perspective rendering algorithms. Therefore, an ideal DTM will provide highly accurate data in the smallest possible storage space.

DTMs developed from maps and imagery are generally stored either in a grid format or as a triangulated irregular network (TIN). Figure 1 shows how a simple contour map might be converted to these two formats.

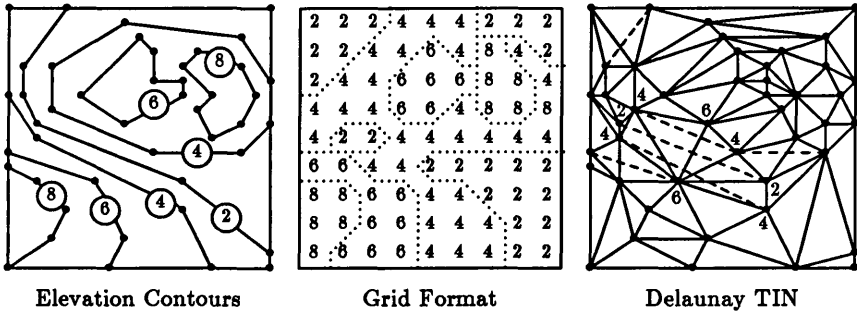


Figure 1. DTMs created with Grid Sampling and Delaunay Triangulation

Grid Sampling

Terrain in a grid format is represented by evenly spaced elevation samples or posts. DTMs produced by the U.S. Geological Survey and Defense Mapping Agency are of this form. Although a grid structure is easy to manipulate, it is limited by its dependence on the sample rate. For example, widely spaced elevation posts may miss important terrain features, producing an inaccurate model. This is clearly demonstrated by comparing DTMs of the same area created with different sample rates as shown in Figure 8 and Figure 9. It is for this reason that recent studies (such as U.S.A.E.T.L. 1987) call for the generation of data with 10 meter resolution or better.

Contrarily, narrow spacing between posts greatly increases the data volume, sometimes unnecessarily. For example, samples taken over a relatively flat area are translated into many polygonal facets when only a few are necessary. This can greatly increase the cost of computer image generation. Figure 1 shows how a grid structure can both contain many more samples than necessary in one area, yet miss important terrain features in another area.

Triangulated Irregular Networks

TINs, on the other hand, contain only those points which significantly affect the topology of the terrain. Included are points along contour lines or ridge and valley breaklines, as well as peaks and pits. These points are linked by edges that define a network of triangular facets conforming to the terrain. TIN structures are widely used in commercial geographic information systems because high resolution may be achieved with relatively little data. For example, flat or smoothly sloped areas may be represented by only a few points, whereas the fine detail in rocky or irregular areas may contain a great number of points. TIN structures are also convenient for generating perspective views, as many rendering algorithms work best with triangles.

Numerous methods have been developed to generate TINs (for example Christian- sen 1978, Dennehy 1982, DeFloriani 1984, Watson 1984, Preparata 1985, Christensen 1987, Correc 1987, Dwyer 1987, McKenna 1987). Most of these algo- rithms are based on Delaunay triangulation, which forms connections between nearest neighbors within a scattered set of isolated elevation posts (Preparata 1985). Extensions to this algorithm typically increase time efficiency, simplify the steps in the algorithm, or add further structure to the TIN. However, Delaunay triangulation has one major drawback. It ignores the natural connections between points along contours and breaklines. These line segments are commonly used in

maps to describe critical terrain features, and are frequently as important as the points themselves. As noted by Christensen (1987) and shown in Figure 1, lines produced with Delaunay triangulation can cross these important breaklines, creating triangular plateaus over or under important features. This results in a misleading and inaccurate DTM.

The cartographic community has recognized this problem and proposed solutions which utilize connections between points along contour lines. However, these algorithms suffer because they require extensive human interaction (Christiansen 1978), do not extend well to handle complex terrain (Dennehy 1982), or double the number of data points as a side effect (Christensen 1987). In addition, none of these algorithms can triangulate intersecting breaklines such as ridge and valley lines and contour lines merged with other breaklines.

THE NEW TRIANGULATION

Our problem was to devise a way of building accurate, yet compact DTMs from common input sources. These sources include contour maps and ridge and valley lines traced from stereo imagery. This data would be input as a series of connected points forming closed polygons, open curves, connected graph structures, and a few isolated points. Our algorithm had to conform to these lines without adding data points.

The resulting algorithm, presented here, is an extension of Fournier and Montuno's 1984 triangulation algorithm developed for simple polygons with non-intersecting edges. This algorithm was an attractive basis for a solution to the more general terrain problem because

1. the polygon edges are inherently part of the triangulation, contributing at least one edge to each triangular facet
2. only the original points defining the polygon are used, resulting in a small data set
3. the algorithm is shown to run in $O(n \log n)$ time with a method that is relatively simple to understand and implement.

Although there is an excellent academic treatise, the Fournier and Montuno triangulation algorithm required a great deal of alteration before it could triangulate diverse terrain data. We created a valuable tool for terrain modelling applications by extending the algorithm to handle the following:

1. Any number of lines and points may be input. This makes the algorithm general enough to triangulate the data from any topological map.
2. Input points may be linked to form closed polygons, open chains of line segments, or even complex graph structures with several edges emanating from each point. This covers all combinations of contour lines, ridge and valley lines, and other breaklines which may occur in a topographic representation. Because maps often contain isolated peak elevations, isolated points may also be included.
3. Lines may connect points that have either a constant elevation (as do contours) or variable elevations (common with breaklines).

The algorithm takes the following steps. First, the data is sorted on point positions. Second, the data set is decomposed into a series of trapezoids. Third, these trapezoids are split by new edges linking points on the trapezoids. Finally, all of

the resulting edges, new and old, are used to define the triangular mesh. The remainder of this section describes these steps in detail.

Setup

Digitized contour lines and/or breaklines are input as lists of connected 3-D coordinates. An edge list is maintained so that a point with multiple references may be condensed to a single point reference with multiple connections. Naturally, if two edges intersect, a point must be placed at the intersection. The only restriction on the data is that elevations for repeated points must agree.

Two steps must be taken before triangulation. First, the points are sorted on their Y value, from bottom to top. Points with the same Y value are sorted on their X value, left to right.

Next, the points must be bounded by a single closed polygon. Although the bounding polygon can be any shape, we selected a rectangle for convenience. We determine the bounding rectangle for the data set, add points at the corners of that rectangle (if necessary), and add edges connecting points along the periphery of that rectangle. New points are assigned elevations which are the weighted averages of their neighbors' elevations, where neighbors are adjacent points connected by an edge.

Defining the Trapezoids

Decomposition of the data into trapezoids parallels the definitions and steps outlined by Fournier and Montuno (1984). As shown in Figure 2, a trapezoid is defined as a four-sided figure with its top and bottom edges parallel to the X axis. These imaginary top and bottom edges each pass through one of the input data points. The side edges, left and right, come from the collection of connecting lines, which are defined by two endpoints each. Therefore, the program stores each trapezoid as a list of six point indices, even though some of the indices may be repeated. For example, Figure 2 also shows that some trapezoids may look very much like triangles. In this case, the top point is also listed as a point on the left edge and a point on the right edge.

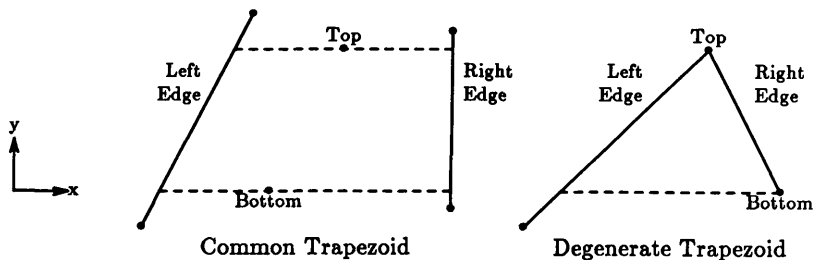


Figure 2. Trapezoids

Trapezoids are developed in the following manner. First, an active list of incomplete trapezoids is initialized. The trapezoids in the active list each have a bottom and two side edges, but no top edge. Initially, every active trapezoid has a bottom edge with the minimal Y value and side edges which are connections to points with the minimal Y value.

Then each data point is examined once, in its sorted order. This point is used to complete active trapezoids and contribute to new trapezoids. First, the active list is searched for trapezoids which have side edges that either surround the current point or belong to that point's edge list. These trapezoids are said to be completed by the current point. Typically, a point with m edges extending downward will complete $m+1$ trapezoids, as shown in Figure 3. Thus, an unconnected point will complete one trapezoid. Once a trapezoid is completed, it is removed from the active list and placed on a separate trapezoid list, with the current point given as its top point. The data set is completely decomposed into trapezoids when all of the vertices have been examined.

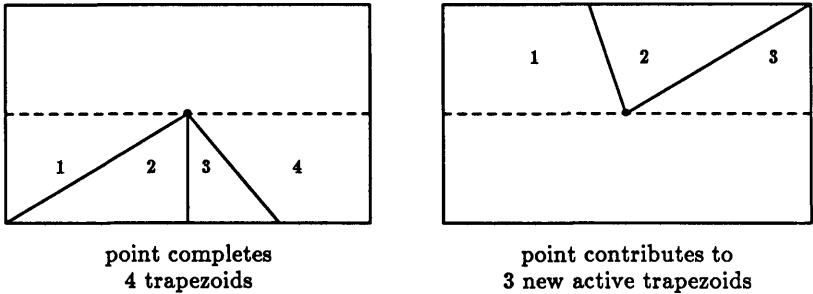


Figure 3. How points contribute to trapezoids

Second, new active trapezoids are created and inserted to the list where the old ones resided. As shown in Figure 3, a point with n edges extending upward contributes to $n+1$ new active trapezoids. The current point is the bottom point of each new active trapezoid.

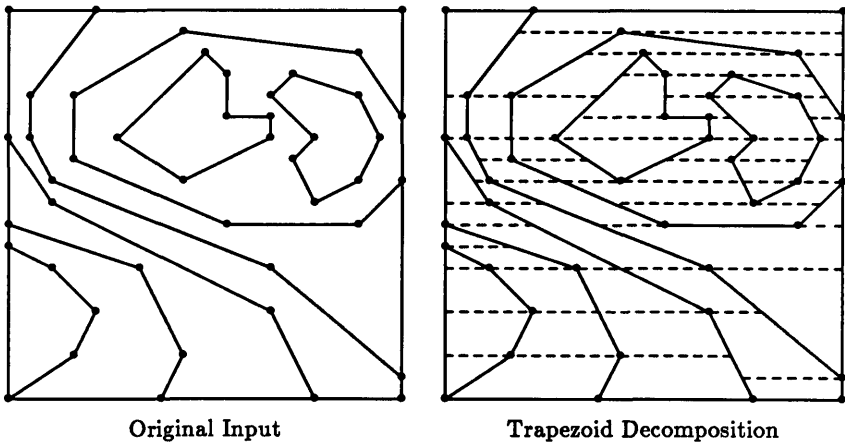


Figure 4. Contours decomposed into trapezoids

Figure 4 shows how the elevation contours from Figure 1 would be decomposed into trapezoids.

Finding the Remaining Triangle Edges

Once all the trapezoids have been found, they are split into triangles by connecting points that lie on opposing sides of the trapezoids. These splitting edges are added to the edge lists of the newly connected points. Each trapezoid that is completely split into triangles is removed from the trapezoid list. When the trapezoid list is empty, all necessary edges have been found. These new edges, combined with the original contour edges, will form the triangular network covering the terrain.

Trapezoids are split in two passes. The first pass recursively splits individual trapezoids. If two or more points lie on the top or bottom edge of the trapezoid, then these points are linked by new edges, added to the points' edge lists. Any point appearing on the top or bottom edge may then serve as the top or bottom point. If a trapezoid has a top point and bottom point which do not lie on the same side edge, then these points are also connected with a new edge. This splits the original trapezoid into two new trapezoids, each of which are examined for further splits. Once a trapezoid becomes triangular in shape, and no more splits are possible, it is removed from the list. Figure 5 shows how the trapezoids from Figure 4 would be split in the first pass.

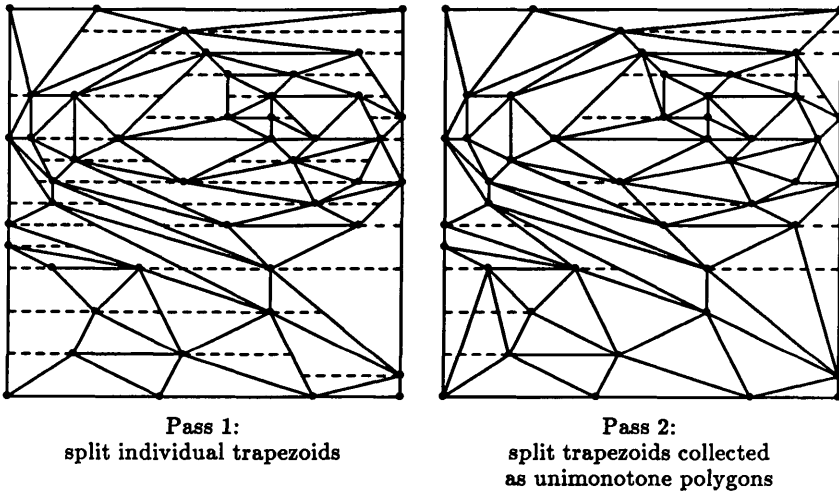


Figure 5. Splitting trapezoids

After the first pass, all remaining trapezoids on the list may be collected to form one or more unimonotone polygons (Fournier 1984). A unimonotone polygon is characterized by a single major edge and a two or more minor edges. The major edge is defined by two endpoints which have the maximum and minimum Y values for the polygon. All points on the minor edges fall within this vertical range. Therefore, all remaining trapezoids with the same major edge are part of the same unimonotone polygon.

The second pass splits the unimonotone polygons into triangles. Two approaches may be taken to decompose each unimonotone polygon into triangles.

1. If a point on a minor edge is linked to the major edge, a triangle may be formed by connecting its minor neighbor to its major neighbor. The new edge must fall within the unimonotone polygon, and therefore the angle formed by the two pre-existing edges must be checked.
2. The two minor neighbors of a minor point may be linked by an edge if that edge falls within the unimonotone polygon.

Figure 5 also shows how the remaining trapezoids would be split in the second pass.

Collecting Edges to Build Triangles

The edges splitting the trapezoids, along with the original edges, form a triangular network covering the area within the original bounding polygon. In the final step, these edges are organized to form a triangle list. This is done by repeating the following steps for each vertex.

1. The current point's connecting edges are sorted in counter-clockwise order.
2. Each pair of adjacent edges on the sorted list form a triangle, because the points at the ends of those edges are linked to one another. Add this triangle to the list only if the indices of the two end points are greater than the index of the current point. Otherwise, this triangle is already on the list.

IMPLEMENTATION

To test our algorithm, we traced contour lines from the field map shown in Figure 6 and fed them to our triangulation routine. This small test area covers 304x482 meters, and is represented by 2014 points. Our routine triangulated this data in 4.85 CPU seconds on a VAX/8530, with 0.35 seconds of that time spent on the initial sorting. The resulting DTM contains 2018 points on 4020 triangles. Figure 7 shows a perspective view of this data.

For comparison, we also generated a grid format DTM from the same contour map. This data was sampled at a regular post spacing of 2 meters and placed in a grid structure containing 152x241 points, or 72480 triangles. Figure 8 shows the resulting grid rendered in perspective. Although this produces an equally accurate representation of the scene, it also contains 18 times the number of points and triangles contained in the TIN. We also sampled the data with 8.25 meters between posts to produce a grid data set with approximately the same number of points and triangles as the TIN. Figure 9 shows the lower resolution data set rendered in perspective. Notice how much detail is lost in the lower resolution.

TABLE 1. Triangular vs. Grid DTMs

DTM	No. of Points	No. of Polygons	Render Time (min.)
Triangulated Contours	2018	4020	2:46.04
2m posts	36,632	72,480	3:52.14
8m posts	2088	3990	2:36.66

Table 1 summarizes our results. Rendering time is measured as minutes of CPU time on a VAX/8530. It is important to note that this sample area is small relative to the coverage required by most applications. Thus, the savings incurred by the triangulation will be greater as the area of interest grows.

CONCLUSION

At a time when demands for highly detailed data over large areas is placing a strain computer systems, we have demonstrated that TINs can represent details far more compactly than a grid format. Furthermore, we have presented a proven algorithm that produces a DTM from critical line features in a reasonable amount of time. This algorithm is superior to previous algorithms because it 1) maintains the integrity of the critical lines, 2) triangulates any series of input lines, including intersecting lines, without requiring any human intervention, and 3) adds no more than 4 points to the original data set.

ACKNOWLEDGEMENTS

I would like to thank J. Mendelson, R. Kelly, H. Tesser, and G. Gardner, without whom this research would not have been possible.

REFERENCES

- Christensen, A.H.J., 1987. Fitting a triangulation to contour lines, *Proceedings of AUTO-CARTO 8*, 57-67.
- Christiansen, H.N. and Sederberg, T.W., 1978. Conversion of complex contour line definition into polygonal element mosaics, *Proceedings of SIGGRAPH '78*, 187-192.
- Correc, Y. and Chapuis, E., 1987. Fast computation of Delaunay triangulations, *Advances in Engineering Software*, 9(2), 77-83.
- DeFloriani, L., Falcidieno, B., Nagy, G., and Pienovi, C., 1984. A hierarchical structure for surface approximation, *Computers and Graphics*, 8(2), 183 - 193.
- Dennehy, T.G., and Ganapathy, S., 1982. A new general triangulation method for planar contours, *Proceedings of SIGGRAPH '82*, 69-74.
- Dwyer, R.A., 1987. Faster divide-and-conquer algorithm for constructing Delaunay triangulations, *Algorithmica*, 2(2), 137-151.
- Fournier, A., and Montuno, D., 1984. Triangulating simple polygons and equivalent problems, *ACM Transactions on Graphics*, 3(2), 153 - 174.
- McKenna, D.G., 1987. The inward spiral method: an improved TIN generation technique and data structure for land planning applications, *Proceedings of AUTO-CARTO 8*, 670-679.
- Preparata, F.P. and Shamos, M.I., 1985. *Computational Geometry*, Springer-Verlag, New York.
- U.S. Army Engineer Topographic Laboratories, 1987. Digital terrain data requirements, *Army Environmental Sciences*, U.S. Army Corps of Engineers, 5(3), 10-11.
- Watson, D.F. and Philip, G.M., 1984. Survey: systematic triangulations, *Computer Vision, Graphics, and Image Processing*, 26, 217-223.

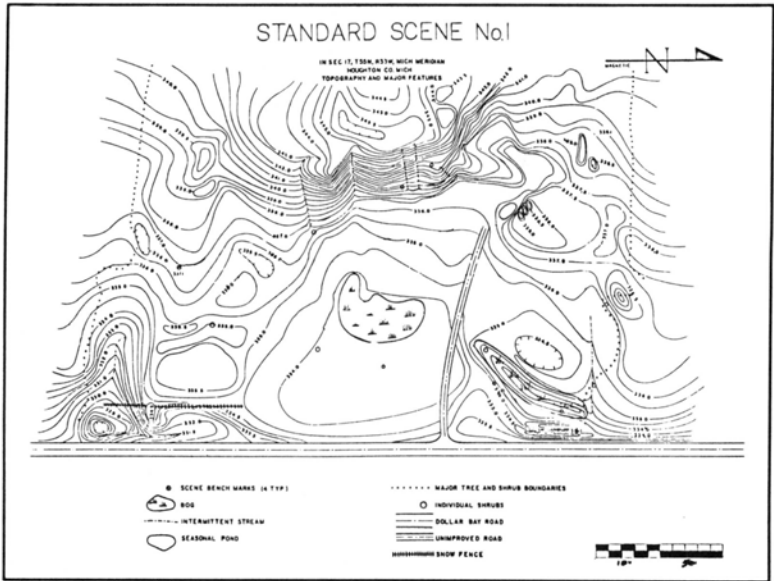


Figure 6. Field map used as input for triangulation

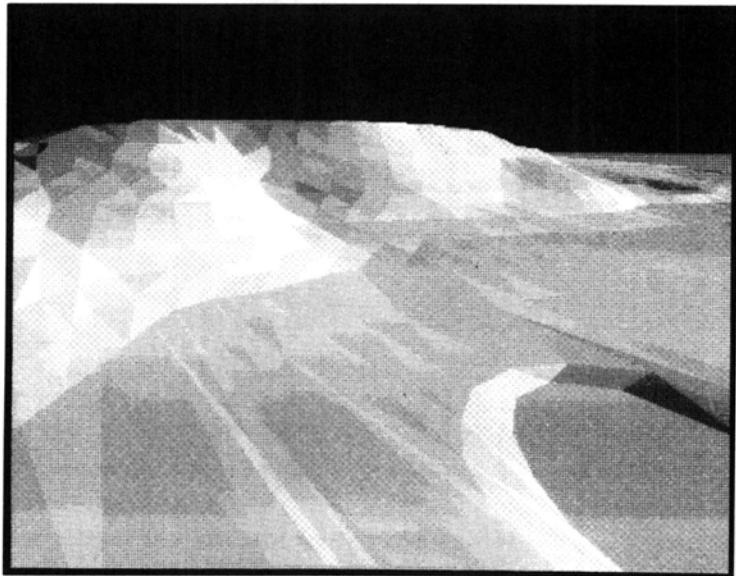


Figure 7. Triangulated field map

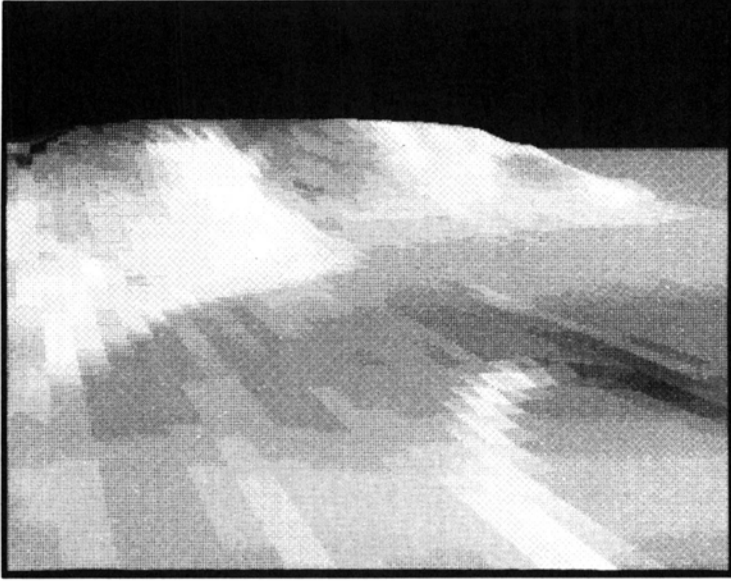


Figure 8. Grid data, sampled every 2 meters

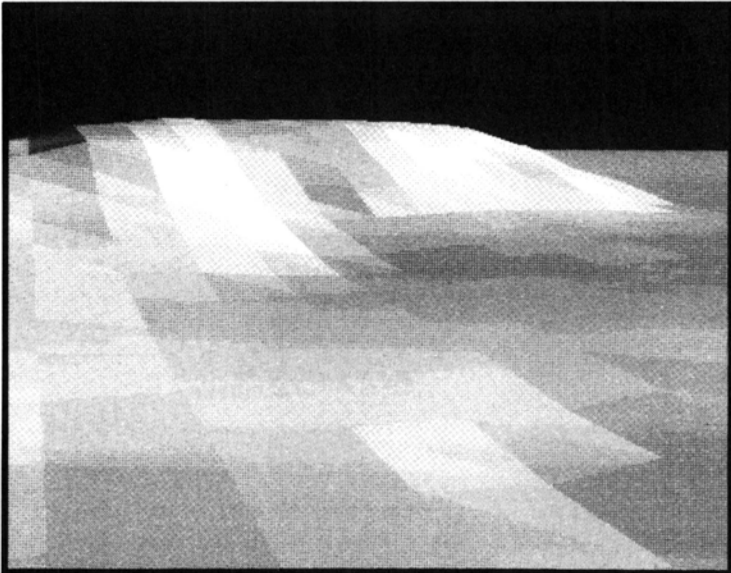


Figure 9. Grid data, sampled every 8.25 meters

A SHORTEST PATH METHOD FOR HIERARCHICAL TERRAIN MODELS *

Renato Barrera
NCGIA University of Maine
120 Boardman Hall
Orono, ME 04469
e_mail RENATO@MECAN1.bitnet

Jesús Vázquez-Gómez
Department of Computer Science
UAM-Atzacapotzalco, México

Abstract

An algorithm is presented for obtaining the shortest path between two points on a terrain represented by a triangular-faced polyhedron. The terrain model is hierarchical, i.e. it has several levels of precision, the representation at each level refining the previous one.

The proposed algorithm consists of two phases. In the initial phase, terrain representations at increasing precision levels are searched for regions where no optimal paths can trespass; these regions are not to be considered any further. In the final phase, a standard shortest path algorithm is applied on the remaining areas.

1 Introduction

A renewal of interest in path finding problems has occurred recently, spurred by several circumstances: a greater number of scientists in geometrical and combinatorial problems, the appearance of inexpensive and powerful computers, and the coming of age of germane applications in robotics, in navigation, in CAD-CAM, etc.

This work deals with efficiently obtaining shortest paths on a triangulated terrain model with v vertices. Were the paths restricted to traversing the terrain through the triangle's edges and vertices, Dijkstra algorithm [AHO 74] would yield the solution in $(v \log(v))$ time. Unfortunately, this approach does not necessarily render an optimal path. The best available exact solution, due to D. Mount [MOUN85], solves this problem in $O(v^2 \log(v))$ time.

A computational time proportional to the square of the number of datapoints is not entirely satisfactory. To that effect, efficient approximate methods that circumvent that difficulty have been devised [SMIT87]. Although fast approximations are very convenient, it is also suitable to consider the improvement of exact path finding methods. This work describes an heuristic that prunes the search region prior to the application an exact optimization algorithm. Our heuristic considers a hierarchical terrain model and a staged elimination of unnecessary regions, each stage corresponding to a level in the hierarchy.

Reduction stages use a branch-and-bound procedure. The steps involved are:

Branch All remaining triangles are subdivided; afterwards, a reasonable short source-destination path on those triangles is generated. The length of such path is an upper bound on the shortest distance, and will be called u .

Bound Let ϵ be an edge in the triangulation. Two functions $S_\epsilon(p)$, $D_\epsilon(p)$ defined for points $p \in \epsilon$ are computed; these functions bound from below the distance of p to the source and destination respectively. All regions circumscribed by polygons whose points obey $S_\epsilon(p) + D_\epsilon(p) > u$ can be discarded, for no shortest route can cross them.

*This work was developed while the authors were at the Electrical Engng. Dept. at CINVESTAV-IPN, México. The second author was supported by a COSNET scholarship.

This paper is organized as follows: Section 2 reviews the terrain model used; Section 3 introduces Mount's algorithm while Section 4 will explain our method. Finally, the last section presents an experimental result and conclusions.

2 Terrain Models

The model used here considers a sequence of $n + 1$ triangulations $\{T_0\}, \dots, \{T_n\}$, as in [BARR87]. Each triangulation is a *refinement* of its predecessors, i.e. for $0 \leq k \leq n$, all vertices of $\{T_k\}$ belong to the set of vertices of $\{T_{k+1}\}$ and each triangle of $\{T_{k+1}\}$ is contained in a single triangle of $\{T_k\}$.

Any triangulation $\{T_i\}$ obeys two conditions:

- i) Its vertices, for whom altitudes are available, form a square grid of $(2^i + 1) \times (2^i + 1)$ points.
- ii) A triangulation similar to the ones in Fig. 1 is induced on its vertices

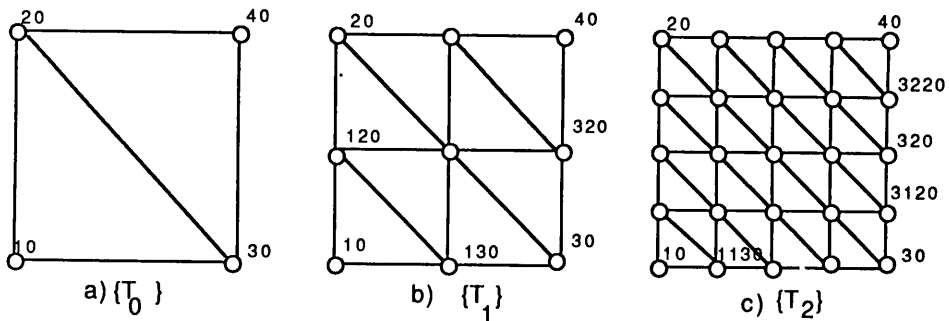


Figure 1: Representation of Triangulations

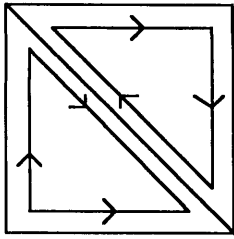
Also, at the finest level $\{T_n\}$, the altitude of any point in the terrain must be accurately described by a linear interpolation of those of the vertices of the triangle containing that point.

2.1 Positional keys for vertices and edges

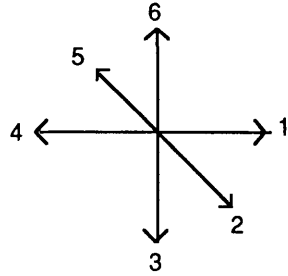
A *positional key* that consists of a string of symbols of the alphabet $\{1, 2, 3, 4\}$ terminated by a symbol $\{0\}$ is assigned to each point in the original $(2^n + 1) \times (2^n + 1)$ array. In our model, the positional key of a point is obtained by a Morton encoding (a.k.a bit interleaving) of its coordinates. Fig. 1.a, 1.b, 1.c show three levels of refinement with some of its corresponding keys.

Since there is a one-to-one correspondence between positional keys and coordinates, the terrain model can be stored as an ordered list of pairs of $\{\text{key}, \text{altitude}\}$. The manner in which two keys are compared is also relevant: if the comparison is lexicographical (i.e. $1230 < 110$) the points are ordered following a Peano curve; if the comparison is numerical (i.e. $110 < 1230$) the points are stored by levels of refinement: first $\{T_0\}$, then $\{T_1\}$, etc.

The edges of triangles in a model $\{T_k\}$ will be assigned a positional key as well. In order to exploit the *monotonicity property*, to be introduced in Section 3, triangles will be oriented (Fig 2.a); i.e. an edge will have a separate version or "directed edge" for each triangle it bounds. Since our model is made of isosceles rectangular triangles, directed edges can have only six directions, coded as shown in Fig 2.b. Therefore, both vertices and directed edges will be encoded in a single way prefixing the positional key of the initial node with the direction code



a) Two copies of an edge



b) Six possible directions of a directed edge

Figure 2: Oriented Edges

(a single node is supposed to have direction '0'). This unifying method of encoding is called an *ϵ -code*.

Considering a terrain model as a hierarchy of refinements and employing a uniform coding schema for vertices and edges is advantageous: their usage favors data compression and provides our procedures with efficient search algorithms.

3 Method of Mount

The (unique) shortest path between two points on a plane is the straight line that joins them. This result has counterparts for the case of polyhedra, both convex and non-convex.

In the remainder of the work it will be assumed that the faces of the polyhedron are triangles, and that both the source s and the destination d are vertices of them. Unless otherwise stated, all paths will start at s and end at d .

This section initially presents necessary conditions for shortest paths, first for convex polyhedra and afterwards for the non-convex case. Finally it will sketch an exact shortest path algorithm due to D. Mount [MOUN85].

The concept of a *planar unfolding* is needed to proceed with the presentation. Let $F_1 \dots F_m$ be the sequence of faces traversed by a path. As said above, s and d are vertices of F_1 and F_m respectively. A series of affine rotations A_1, \dots, A_m can be applied to the faces so that:

- $A_k(F_k)$ is on the (x, y) plane
- If e_k is the common edge between the faces F_k and F_{k+1} , then $A_k(e_k)$ and $A_{k+1}(e_k)$ coincide.

The sequence of faces $A_1(F_1) \dots A_m(F_m)$ for a given path is called its *planar unfolding*, since after the transformations the path will lie flat on the horizontal surface.

The following necessary condition can be stated using unfoldings:

Condition I In a convex polyhedron, the image of an optimal path on its planar unfolding must be a straight line.

Figure 3 shows a pyramid, a shortest path and its unfolding.

Condition I is not sufficient: if the bottom of the pyramid in fig 3.a is ignored, the paths between the extreme points might cross two possible sequences of faces: $F_6 F_5 F_4 F_3$ and $F_1 F_2$. The planar unfoldings of both sequences admit a straight line between source and destination (Fig 3.b). The first path is truly optimal. The second one is a local optimum, i.e, best among of all possible paths through that sequence of faces. Cases can be found where several optimal paths exist.

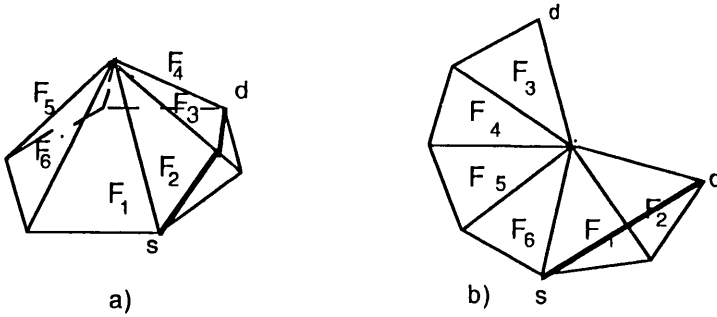


Figure 3. A shortest path on a Convex Polyhedron and its unfolding

In the case of a non-convex polyhedron, the mapping of an optimal path on its planar unfolding will not necessarily render a straight line. Condition I becomes :

Condition II In a non-convex polyhedron, the image of an optimal path on its planar unfolding must be composed of one or more straight line segments; any two consecutive segments must be joined at a vertex of a face and the angle between them must be at least 180 degrees.

Fig 4 illustrates that condition, which again is only necessary for the optimality of a path. Paths obeying it are called geodesics; they can be proved to be locally optimal, i.e. all detours inside the planar unfolding render longer distances.

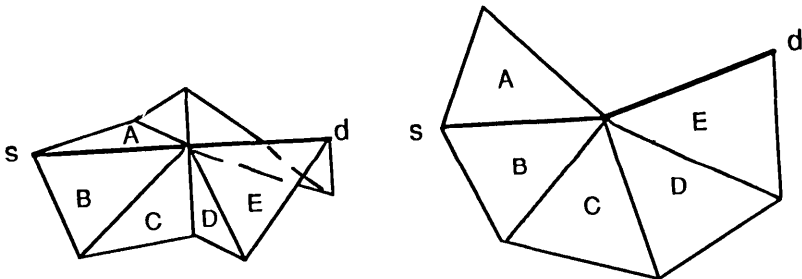


Figure 4: Planar Unfolding on a nonconvex polyhedron

Geodesics are interesting: they are uniquely specified by the sequence of its traversed faces (called "history") and the optimal paths are counted among their numbers.

The generation of histories is straightforward: Fig 5.a shows a polyhedron, and Fig 5.b the connectivity graph of its faces. Fig 5.c displays a tree whose nodes are labelled with the names of faces: Its root with A and its leaves with D. All possible histories that start with A and end with D correspond to paths between the root and a leaf of 5.c. Any initial segment of an history will be called a partial history.

The generation of histories is similar to the exploration by gradual expansion of all possible paths in a graph. Thus, the process of generating Fig 5.c from the data of Fig 5.b starts with A as the only partial history. Two histories AB and AD are generated by the extension of A; AD has reached the destination and will not be further extended, so AB is extended to render

two partial histories ABC' and ABG , which in turn are extended to $ABCD$, $ABC'F$, $ABGF$, etc

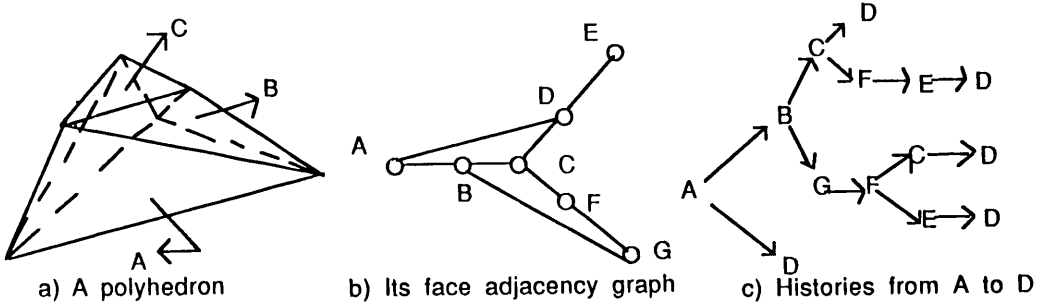


Figure 5: Faces and histories

Two things must be noted in the example of fig 5.c:

- An edge between two faces can be crossed in both directions (e.g. in Fig 5.c the edge common to C' and F belongs to $ABC'FED$ and $ABGFCD$). Considering an edge as made of two directed copies or d_edges simplifies greatly the computations. Thus, the edge common to C' and F will have two copies: a d_edge to go from C' to F , and another one to go from F to C' . As shown in Section 2, directed edges can be easily accessed in our model by means of its e_code.
- A d_edge might be found in more than one history (as $C'D$, found in $ABC'D$ and $ABGFCD$)

The characterization of geodesics by means of histories drastically reduces the number of paths considered for optimality, and it suggests a method based on the generation of histories. It presupposes the existence of a heap of histories ordered by the shortest distance of its shortest geodesic, of a floating point number D that measures the shortest geodesic found so far, and a function $distance(history)$ that renders the length of the shortest geodesic inside that history.

The algorithm has the following steps:

- Push all faces neighboring the source into the heap. $D = \infty$
- $history = pop(heap)$
- if $distance(history) > D$, terminate the algorithm. The shortest geodesic found so far is an optimal path.
- If $history$ has reached the destination, make D the minimum between the old D and $distance(history)$. Else extend $history$ and push all its descendants into the heap.
- Go to i)

The algorithm in [MOUN85] is based on those ideas. Its implementation employs a construction called "wedge", that resumes a partial history. A new wedge is generated every time a history is extended and collides with a new directed edge. Among other information, a wedge contains

- The address of the directed edge.
- The wedge's basis, i.e. a characterization of the set of points on the edge reachable by geodesics inside a planar unfolding.
- A function that renders the distance from all points on the basis to s .
- A pointer to the previous wedge in the history

The cited algorithm considers directed edges. This artifice makes the problem monotonic, i.e. makes all edges traversable in only one way. It can be proved that monotonicity guarantees the connectivity of the basis of a wedge, thus greatly simplifying the computations. The distance from a point p on a wedge's basis to s is given by a circle, that is, by a formula of the type $(p_x - r_0)^2 + (p_y - \eta_0)^2 + c$, where p_x, p_y are coordinates of p on the planar unfolding. The number of wedges that can coexist on a given directed edge is proportional to the total number of vertices. Since the number of directed edges is proportional to that of vertices, $O(v^2)$ wedges might be generated and the algorithm needs $O(v^2)$ memory. Considering that the wedges have to be kept on a heap gives a computational complexity of $O(v^2 \log(v))$.

The algorithm has been only sketched, and many features related to the extension of a wedge have been omitted; e.g. how a wedge might be split, narrowed or turned around a vertex. The basis of two wedges can also collide on a directed edge; in that case at least one of the basis must be narrowed so as to keep them disjoint.

4 Proposed Method

Existing optimal algorithms for the shortest route problem require a $O(v^2 \log(v))$ computational time. This means that the amount of computation grows with the square of the number of vertices, and with the fourth power of the precision.

Those figures tempt the user to decrease as much as possible the number of points under consideration and to perform the reduction at coarse scales.

That seems generally feasible. Even though there are times when an optimal solution occupies all triangles, in most of the cases only a fraction of the terrain is traversed by an optimal route. The greater percentage of the computation time is spent in fruitlessly exploring regions in which no optimal solution can exist.

Thus, a preprocessing method that reduces the area under exploration is indicated. The amount of preprocessing should be further diminished if that method is successively applied on increasing finer terrain models.

The procedure selected is one of the branch and bound type. The algorithm has the following steps:

- i) Select the coarsest terrain model as the work triangulation $\{T_w\}$. Initially, all of its triangles are available.
- ii) Obtain a *good* path that traverses only those triangles available from $\{T_w\}$. Let u be its length.
- iii) For all points p on the boundary of the triangles of $\{T_w\}$ obtain a lower bound on the distance $S(p)$ (or $D(p)$) to s (or d)
- iv) Discard all triangles enclosed by a cycle of directed edges obeying $S(p) + D(p) \geq u$
- v) If the triangulation is already at its finest level, apply an exact algorithm, otherwise make $\{T_w\}$ a refinement of the available triangles, and go to step i).

An example of the application of those ideas is the following:

Suppose that s and d are located at $(-l, 0, 0), (l, 0, 0)$. Then, for all points p on the terrain, $S(p)$ and $D(p)$ can be given by the distance of the horizontal projection of p to s and d respectively. If a path of length u is available, then no optimal path can go outside the ellipse whose locus is given by $2(x/u)^2 + 2(y)^2/(u^2 - l^2) = 1$

All triangles outside the ellipse can be discarded. Fig. 5 illustrates this concept.

The elliptic formula gives a good initial reduction, but can only work on flat surfaces. In order to proceed to better approximations, a method to obtain $S(p)$ and $D(p)$ was developed that simplifies some features in an exact algorithm.

In an exact algorithm:

- The distance to s from points on the base of a wedge is given by a circle.

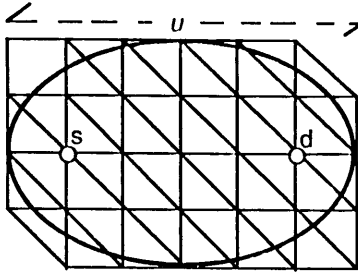


Figure 6: The elliptical approximation

- $O(r)$ wedges with mutually exclusive basis can coexist on an edge.
- The wedge closest to s is expanded, making it possible for $O(r)$ wedges to be expanded on a single edge.

In the algorithm for obtaining lower bounds, a construction called *approximated wedge* (ap_wedge) will be used. For any given directed edge:

- The distance from points in an ap_wedge base to the s or to d is given by a circle.
- Up to a constant number of ap_wedges can coexist on an edge.
- Only one ap_wedge can be expanded per edge. Several ap_wedges can coexist on an edge prior to expansion. When expansion is to be performed, a subtending ap_wedge, i.e. one whose distance to the origin is not greater than any of the existing ap_wedges, should be obtained.

Lower bounds $S()$, $D()$ are obtained by the following algorithm, that consider only those non-discarded edges:

- Obtain ap_wedges for the edges neighboring s (or d). Expand them. Push their expansions into the heap.
- If the heap is empty, terminate.
- $wedge = pop(heap)$. Let e be the corresponding edge, $\{W_e\}$ be the set of wedges residing on e . Complement $\{W_e\}$ with the expansion of those ap_wedges that have not affected it yet.
- Let $wedge'$ be a wedge subtending $\{W_e\}$. Expand $wedge'$, and push into the heap those new ap_wedges that are not incident on an expanded edge.
- go to ii)

The previous method for evaluating $S(p)$, $D(p)$ takes care of all those regions enclosed into polygons whose points obey $S(p) + D(p) \geq u$. The respective internal edges will not be expanded and, therefore, their interior regions will not be considered.

5 Example

Fig 7 shows the results of our algorithm in a 9×9 terrain model, with three stages of reduction. Overall terrain reduction shown in Fig. 7.a was 40%.

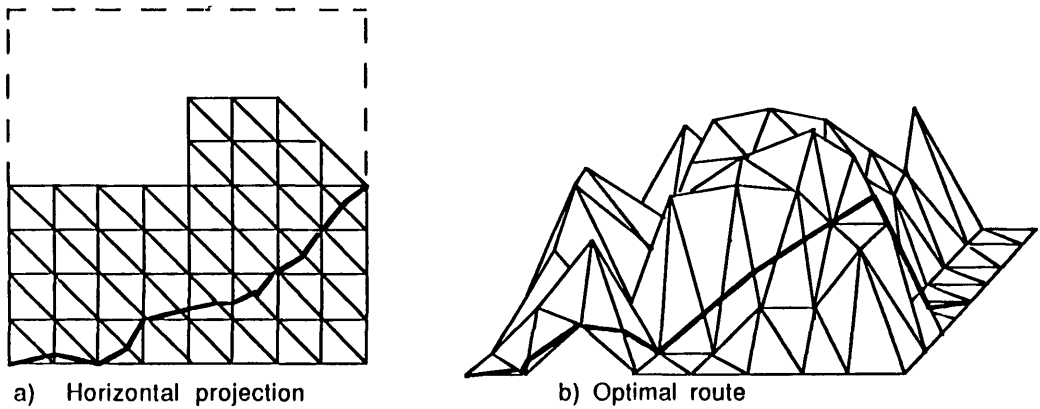


Figure 7: An Example

The computation time was 6 minutes in a PC-XT without floating point accelerator. The exact method lasted 8 minutes.

References

- [AHO 74] A.V.
Aho, J.E. Hopcroft, J.D. Ullman The Design and Analysis of Computer Algorithms. Addison-Wesley, 1974.
- [BARR87] R. Barrera, A. Hinojosa "Compression methods for terrain relief". Intl. Colloquium in Progress in Terrain Modelling, Copenhagen, May 20-22, 1987.
- [MOUN85] D.M. Mount "Voronoi Diagrams on the Surface of a Polyhedron". Rept. CS-TR-1496, Computer Science Technical Report Series, U. of Maryland, College Park MD., May 1985.
- [SMIT87] T.R. Smith, R.E.Parker "An analysis of the efficacy and efficiency of hierarchical procedures for computing trajectories over complex surfaces" *European Journal of Operational Rsch.* Vol 30, pp 327-338, 1987.

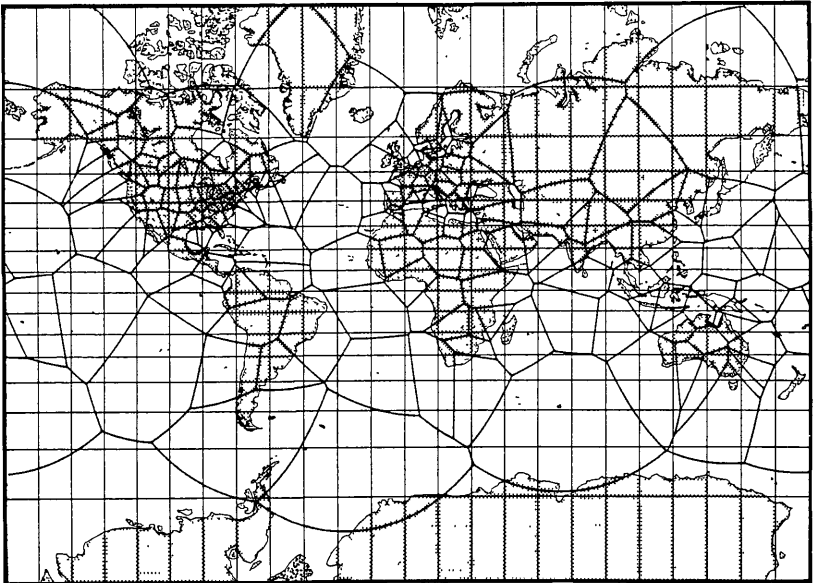
**HIPPARCHUS DATA STRUCTURES:
POINTS, LINES AND REGIONS IN SPHERICAL VORONOI GRID**

Hrvoje Lukatela
2320 Uxbridge Drive, Calgary, AB T2N 3Z6 CANADA
(Envoy 100: lukatela)

ABSTRACT

Hipparchus geo-spatial manager (cf. Auto-Carto/8 introduction paper) is an operational software package that provides geometrical and geo-relational functions to applications that manipulate spatial objects. It is capable of geodetic precision levels, and fully honors the isometric, spheroidal nature of the terrestrial surface and orbit data-space. A Voronoi tessellation is used as a base for its domain partitioning grid. This paper outlines data structures used to represent 0, 1 and 2 dimensional surface objects: sets consisting of discrete points and lines, and non-simply connected regions.

The paper also discusses general characteristics of a family of computational geometry algorithms which evaluate spatial unions and intersections by operating simultaneously on the digital model of the spheroidal Voronoi grid and on the object structures.



Hipparchus Geopositioning Model: An Example of Voronoi Cell Grid

INTRODUCTION

Point, line and region sets represent the most common classes of spatial objects that a geographically-oriented application system must be capable of manipulating in a meaningful way. While the full nature of such manipulations depends on the domain and purpose of the

application itself, spatial unions and intersections represent their reoccurring "building-blocks". It is therefore desirable to provide this functionality in a packaged form, in order to avoid re-programming of identical procedures in many different development projects. This is one of the functions of the Hipparchus software package. (cf. Auto-Carto/8: Hipparchus Geopositioning Model: an Overview, same author. Understanding of the model elements described therein is assumed.) This paper details basic data structures and computational methodology used in this particular functional segment of the package. The implementation follows true ellipsoidal frame of reference; to simplify this presentation, structures will be considered only in their spherical form.

Certain principles apply to all three object classes. Each class represents, conceptually, a pertinently dimensional set of surface points. Each set can consist of a finite number of simple component-sets of the same dimension. (Component-sets are "fragments" of the set in the strict sense of its spatial extent; all non-spatial characteristics describe the whole set or object, none can be specific to a particular component-set.) Since an object modeled by the system can lose completely its spatial extent, the system must not only recognize empty sets, but also be able to use them in union and intersection productions. One and two dimensional sets are numerically represented by finite, ordered sets of vertices, conceptually connected with great circle segments.

All three spatial object classes are commonly used and exchanged by various systems that build and use digital models of geographically distributed data. Assuming that point locations are defined in a coordinate system - appropriate to the particular reference surface - "neutral" form of numerical object representation is usually constructed as a set of ordered point coordinates, with component-sets delineated by a "not-a-coordinate" token. Two consecutive tokens signify absence of further component-sets, i.e. end of point data representing the object. A data-item following the last token identifies the object. Since component-sets of a point set are single point coordinates, its "neutral" representation can be (and commonly is) simplified by discarding the component-set delineation tokens, and terminating the whole set with a single token.

If different point coordinates are represented by Pta, Ptb, Ptc, etc., delineation tokens by *, and if a quote-string data-item is used to identify the object, examples of "neutral" representation of point, line and region objects could be:

Point set:

Pta Ptb Ptc Ptd Pte Ptf Ptg Pth Pti Ptj * "letter boxes"

Line set:

Pta Ptb Ptc * Ptd Pte * Ptf Ptg Pth Pti * Ptj Ptk * * "snow fences"

Non-simply connected region:

Pta Ptb Ptc Ptd Pte Ptf Ptg Pth * Ptl Ptk Ptj Pti * * "Crater Lake"

SPHERICAL VORONOI GRID AND COMMON DATA ITEMS AND STRUCTURES

In addition to the spherical or spheroidal reference surface, Hipparchus spatial index - dual of the Voronoi polygon grid - forms another dominant spatial feature of the system. Ordered polygons

represent a series of identifiable "data-cells". Each surface point belongs to one - and only one - cell. (Points on the edge belong, by convention, to a cell with a lower number.) No restrictions are placed on the relative position of objects and the cell grid: an object can be contained completely inside a single cell, or extend over any number of cells. Consecutive vertices on a line can belong to the same cell, to two neighbour cells, or to two cells which are not neighbors. Linear segment connecting two consecutive points can pass through any number of cells.

Only three abstract data types - in addition to the object identifier - are used in object representation: cell ordinal number, local point coordinate pair and cell-boundary intersection coordinate. (Their mapping into data types intrinsic to a particular computer environment can vary from implementation to implementation). In addition, Hipparchus structures include (unsigned) count of elements in various (ordered) lists and (aggregate) clusters, and locators of their subordinate lists or clusters. (Locators are the only elements which need conversion as the objects are moved from memory to external storage and back.) Implementation specific mechanism exists for manipulation of cell identifier lists, in order to save space and speed up list search algorithms.

Object structures are designed with two - often diverging - objectives in mind: efficiency of transformation between neutral, external and internal object representation and efficiency of evaluation of spatial unions and intersections.

Several structures are used as building-blocks in the representation of spatial objects:

Occupied cell descriptor: Structure describing a cell which contains one or more aggregate (non-ordered) points that belong to the object. The structure consists of:

Cell identifier
Local coordinate cluster length
Local coordinate cluster locator

Line descriptor: Structure describing a line that forms either one among the components of a line set, or one among the boundary rings in a non-simply connected region. The structure consists of:

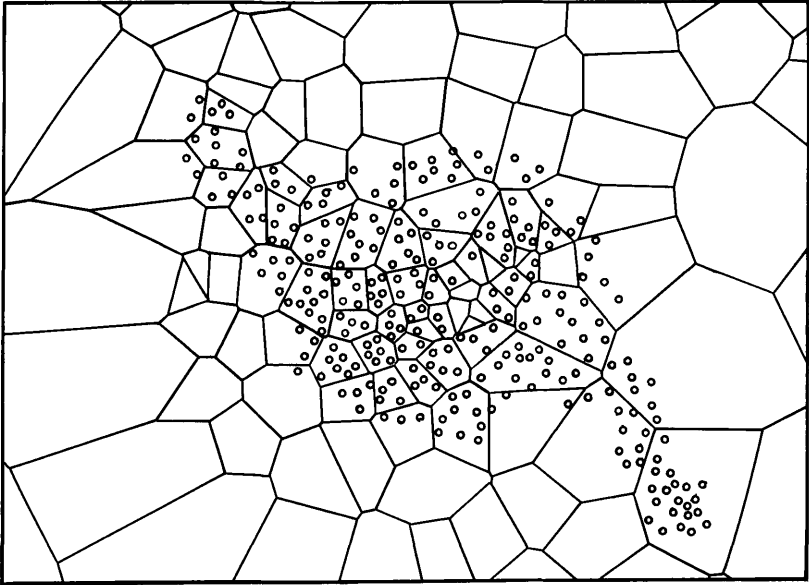
Traversed cell descriptor list length
Traversed cell descriptor list locator

Traversed cell descriptor: Structure describing either that section of a line or the boundary ring that is inside one cell, or the complete line/ring contained in a single cell. The structure consists of:

Cell identifier
Local coordinate list length
Local coordinate list locator
Cell boundary intersection coordinate

In the case where this structure portrays only a section of the line or ring, boundary intersection coordinate defines the point where the line leaves the cell. This item will be void if the whole line or boundary ring is inside a single cell, and, likewise, in the last cell of a line. In the case of single cell boundary ring, coordinate list attached to this structure is circular by definition, the start vertex point is not repeated at the end of the list.

POINT SET



Point set object descriptor: Structure defining a point set object and describing its spatial extent. The structure consists of:

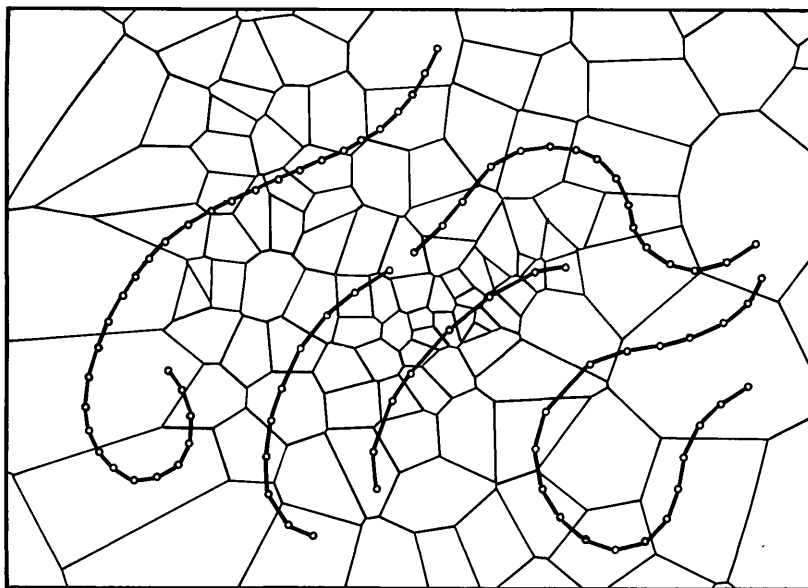
Object identifier
Occupied cell descriptor list length
Occupied cell descriptor list locator

Objects that by definition never exceed single point location (i.e. simple point objects) can be represented by a much simpler, self-contained structure:

Object identifier
Cell identifier
Local point coordinate pair

Since single point component-sets represent parts of the spatial extent of the same object, no two of them must be closer than the minimum spatial resolution that the system is capable of representing.

LINE SET



Line set object descriptor: Structure defining a line set object and describing its spatial extent. The structure consists of:

Object identifier

Traversed cell identifier list

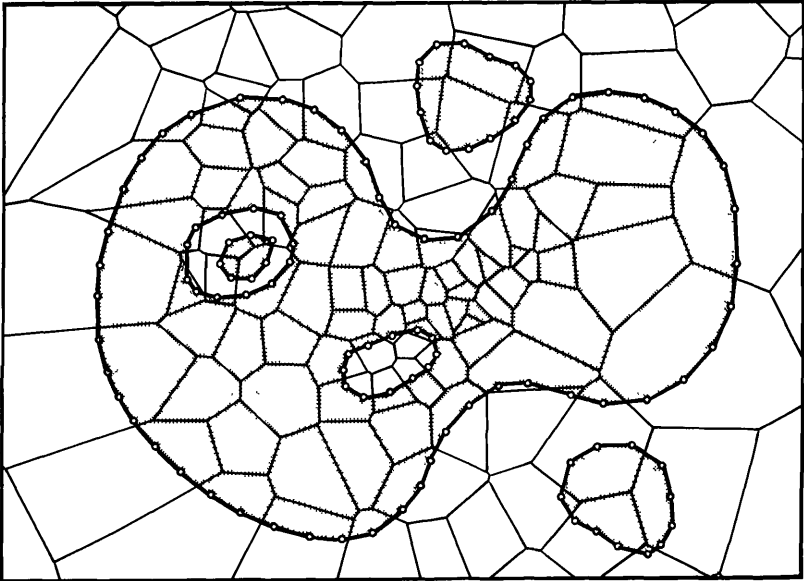
Line descriptor cluster length

Line descriptor cluster locator

Consecutive vertices on the line must not coincide. The package provides the mechanism for the detection of coincident non-consecutive vertices, as well as intersection of linear segments. Such conditions are acceptable; any rules to the contrary must be defined within, and enforced by, the application. As mentioned above, a single segment can span many cells. Traversed cell may, in such case, contain no vertex; a single line fragment in it will be defined by the cell boundary intersection coordinate of two consecutive traversed cell descriptors.

Boundary intersection coordinate is a linear measure. Unlike vertices - which exist in "neutral" representation, and are hence intrinsic to the object - intersection is an artifact of the internal representation. The implementation therefore ensures that the intersection is numerically encoded with greater spatial resolution than that of the vertex.

REGION



Region object descriptor: Structure defining a non-simply connected region object and describing its spatial extent. The structure consists of:

- Object identifier**
- Traversed cell identifier list**
- Interior cell identifier list**
- Line descriptor cluster length**
- Line descriptor cluster locator**

Consecutive vertices must not coincide, the line must not cross itself, and the cell boundary intersection coordinate is treated as explained previously. Boundary ring direction is significant: by a common convention, interior is on the left-hand side.

The structure is capable of representing non-simply connected regions of any width (parallel "islands") or depth ("lake-island-lake..."). The package provides the width- and depth-unrestricted mechanism for the detection of inconsistencies in the ring direction.

While the manipulation of non-simply connected regions increases significantly the complexity of many algorithms employed by the package, this is an indispensable feature: either the union or the intersection of two simply-connected regions will, in general case, be a non-simply connected set. An application lacking this feature (i.e. an application restricted to simply-connected regions) would therefore be unable to treat a result of some of the two-dimensional unions or intersections in the same way as either of the production constituents. This might not be a major problem for applications that produce only graphical displays of such productions, but can weaken significantly those applications that create new spatial objects by applying a combination of spatial and non-spatial operators to the objects existing on their data bases.

Spatial Union and Intersection Algorithms

The critical feature of all union and intersection algorithms in the Hipparchus package is their ability to avoid the spheroidal computational geometry in all instances where the spatial relationships can be resolved by simple comparison of cell identifier lists. As mentioned above, such lists are encoded in a form which exploits the sequences of cell numbers. (Strategic cell number clustering provides therefore an additional mechanism by which the application can improve the efficiency of the package.)

Once the problem requires that the computational geometry be performed, the algorithm will isolate cell or cells to which the solution is restricted. If no proximity criterion is involved - i.e. in case of unions and intersections between union-compatible sets - those cells will always represent the intersection of lists of traversed cells of two constituents. If the production involves a proximity criterion, the minimum and maximum cell-vertex distances in a pair of cells is used to reject from the geometry processing those sections of the object which either can not form part of the solution, or which are known to be part of the solution and can be mapped directly into the output set.

Since, in general case, a cell can contain fractions of more than one component-set, all geometry result elements must be stacked up, and the complete list must be traversed in order to properly connect new line or boundary ring segments. Such traverses are also restricted to the elements occurring within one cell.

Two-dimensional union and intersection algorithms must recognize coincident linear segments, in order to detect collapsing boundaries of two-dimensional objects. Where the complete boundary of both two-dimensional object collapses, the algorithm must be capable of determining whether the resulting object covers the whole domain, or whether it has no spatial coverage at all.

Line component-sets are obviously aggregate, but the boundary rings in a non-simply connected region could be ordered (width- or depth-first), according to their position in the component-set hierarchy. It is interesting that such ordering contributes nothing toward the simplification of union and intersection algorithms, and presents the problem of developing a reasonable convention which defines highest-order component-set on the spheroidal surface. Consequently, boundary rings in the Hipparchus structures are considered to be aggregate.

Finally, all computational geometry in Hipparchus package is performed using global coordinates. This avoids completely the problem of possible topology discrepancies between the spheroid and the projection plane; consequently, the design is free from any restrictions on the maximum length of the segment between consecutive vertices. Direction cosine form of the global coordinates provides for computational geometry algorithms based on vector algebra; such algorithms are both easier to program and simpler to test than the algorithms based on the conventional spherical latitude/longitude coordinates.

INTERACTIVE ANALYTICAL DISPLAYS FOR SPATIAL DECISION SUPPORT SYSTEMS

Marc P. Armstrong
Panagiotis Lolonis

Department of Geography
The University of Iowa
Iowa City, IA 52242
BLAMMGPD@UIAMVS.BITNET

INTRODUCTION

Geographic information system technology is now being placed in the hands of decision-makers who often have little or no cartographic training. Although it is possible that these users will glean insight from the displays that they generate, map readers are able to recover more information from a carefully designed display, than from a poorly designed display. In our application, we are attempting to integrate spatial modelling techniques (e.g. location-allocation models) with the spatial data handling and display capabilities of GIS, to improve the process of spatial decision-making. Although it is important that decision-makers be provided with a capability of viewing the results of spatial models, we wish to isolate users from technical, mundane details of generating the display. To achieve this goal we have developed techniques that allow an expert system to assume responsibility for some aspects of thematic map layout and design.

PREVIOUS WORK

Computer-Assisted Map Design

Monmonier has described a general system used to monitor the process of layout for the National Atlas of the United States (Monmonier, 1982:159). In that system, predetermined configurations are used to guide page design. Each configuration is assigned a code, and pages are built from composites of these codes. Monmonier (1982: 159-163) also describes several other areas of cartographic production that lend themselves to automation: aggregation, data reduction to control display of attribute data, and the general problem of map layout. In his discussion, Monmonier distinguishes movable and fixed map components. In our application, we expand this notion to develop rules for determining the motility of map components.

Broome, Beard, and Martinez (1987) describe an approach to determining the conditions under which an inset map should be produced to enhance legibility, and then to decide where the inset should be placed. The system attempts to mimic the process through which a cartographic designer would arrive at the same answer. The map is divided into cells, and the number of cartographic objects in each cell is tabulated, and smoothed using a 9 cell window. This notion of feature density is important in our application, but in a slightly different way because we are concerned with finding empty areas, rather than areas with high feature density.

Artificial Intelligence Approaches

Several researchers have been involved in exploring the application of artificial intelligence principles to cartographic problem solving.

Maggio (1987) suggests an approach in which a GIS becomes a part of an expert system. Morse (1987) developed an approach in which existing GIS software (MOSS) is augmented by an expert system shell to facilitate the construction of a rule-based forest management system. A general approach to developing an expert system for map design is described by Robinson and Jackson (1985). Follow-up work on a related project by Mackaness and Fisher (1986) shows how some problems of map design can be resolved, and also points out the great complexity involved in the process. Nevertheless, although Fisher and Mackaness (1986) conclude that map design expert systems are possible to construct, they stress that cartographic expertise is both difficult to elicit and specify.

APPROACH TO SYSTEM DEVELOPMENT

In this paper we allocate map components to areas of a viewport in such a way to lead to a maximum, balanced filling of space. To accomplish this task, we use production rules to guide placement of map components (e.g. legends and titles). The production rules are derived from two general sources: cartography textbooks that purport to deal with design, and personal experiences of the authors. From the texts, we incorporated ideas about framing, and allocation of space using the general strategy of "thumbnail sketches" advocated by several authors (Robinson, et al. 1984; Cuff and Mattson, 1982; Dent, 1985). To make the problem manageable each map component is delineated by its extent (bounding rectangle). This approach allows us to deal with the size and location of elements without the excessive amount of computation required for irregularly shaped objects.

We have adopted a three stage approach to system development. The first stage transforms the study area into an abstract representation which can be easily manipulated by declarative programming languages. The second stage arranges the map components within the viewport using a set of rules designed to meet cartographic requirements while avoiding conflict in displaying the map components. The third stage consists of the transformation of the map components from their abstract representation into a displayable format.

Input to the Process

Two types of input are required at this stage. The first is a set of vector chains (Figure 1) which define the perimeter of the area that must not be overlapped by any other map component (e.g. legend). Each straight line segment is represented as a fact in the knowledge base. The second type of input consists of information about the number, size, and shape of the remaining map components (e.g. legend). To restrict the problem, we assume that the minimum space requirements of each component has been determined at an earlier stage of the layout and design process.

Rasterization of Map Boundaries

Vector format data are not well suited for detecting and evaluating the size and shape of empty spaces where map components can be placed. The use of vector data causes an additional problem because our processes are formulated in a declarative environment. In addition, the chains comprising the boundary of the study area might have superfluous sinuosity; cartographers consider only the general shape and not the details of objects when they make decisions about map layout (Cuff and Mattson, 1982, p. 75). Unnecessary detail can also exhibit more virulent effects associated with increased memory requirements and decreased processing speed. For these reasons, we transform chains to a coarse raster format (Figure 1). The size of the grid depends on the range of the coordinate values and the dimensions of the display medium, and should neither introduce unnecessary detail, nor substantially distort the area to be mapped.

When the cell size has been specified, then determining the cells which correspond to the endpoints of each straight line segment is straightforward. Specifically, the row and column number of each endpoint cell on a straight line segment can be computed using the following formulae:

$$Xci = (\text{int} (Xi) \text{DIV} d) + 1 \quad (1)$$

$$Yci = (\text{int} (Yi) \text{DIV} d) + 1 \quad (2)$$

where Xci : the column number of the cell for endpoint i

Yci : the row number of the cell for endpoint i

Xi : the x coordinate of endpoint i

Yi : the y coordinate of endpoint i

d : integer denoting the cell size in the same units as Xi and Yi.

The determination of row and column numbers is iterative, and can be implemented in PROLOG using backtracking and the fail predicate. The results of this process are asserted as facts in the database. Each fact stores the column and row numbers of the endpoints of the corresponding segment. Figure 2 displays how the areas shown in Figure 1 are transformed after the application of this process.

Rasterization of Segments

In order to determine the empty space around the map, all grid cells comprising the boundary of the study area are defined. At this point although we know the grid cells that terminate each straight line segment, the remaining cells along the segment are calculated in a vector to raster conversion. From the various algorithms which rasterize line segments, Bresenham's was chosen because it is efficient and easy to implement. Bresenham's algorithm uses the row and column numbers of the end points of a straight line segment and returns the row and column numbers of the set of grid cells which most closely approximate the line segment (Foley and Van Dam, 1982:432).

We implemented Bresenham's algorithm in PROLOG. The program uses a set of facts which define the endpoints of border segments of the study area, and returns another set of facts which describe the complete set of grid cells bounding the study area. The structure of the new facts is shown in Figure 3, where each fact corresponds to a cell defined by its column (X) and row (Y) number. In addition, each pixel has an attribute indicating whether the corresponding cell is

part of the border of the area to be mapped. All remaining cells are labeled "blank". If, for example, the data in Figure 2 were processed using the algorithm, the outcome would appear as Figure 4.

Layout to Fill the Display Frame

The next steps eliminate empty space surrounding the mapped area, and determine the frame dimensions such that the frame proportionally fits the display area. To achieve this goal it is necessary to determine the extent of the area, which is used as a "core" for any further manipulations (e.g. translation, scaling). If the extent fits in the display frame, and the remaining map components can be arranged in the remaining empty space without violating any cartographic standards, then a satisfactory solution has been identified. Any attempt to further increase the scale of the map would result in crossing the display border and in hiding part of the information from the user. This suggests that it is promising to use the extent of the study area as a starting point for solving the problem of allocating map components in the display.

Given that the area to be mapped has been rasterized, and that the cells defining the border of the area are given as facts (Figure 3), the extent of a picture is determined by scanning all facts which have "pixel" as a functor and "map" as an attribute value and then finding the minimum and maximum row and column number. In Figure 4, for example, the extent is defined by rectangle with corners (2,2) (9,9). Row and column one and ten of that figure do not have shaded cells, and thus can be eliminated without any loss of information (Figure 5). Notice that the remaining rows and columns have been renumbered for convenience. Determining the extent of an object is a double loop iterative process which is implemented in PROLOG by using the fail predicate and backtracking.

After determining the extent of the area, the next step is to place the extent within a frame with dimensions proportional to the dimensions of the viewport. If the dimensions of the extent are not proportional to the viewport, columns or rows must be added to the extent to achieve proportionality. The resulting frame is called the adjusted extent. If required, then the number of columns that need to be added can be computed using the formula:

$$X = (L / W) * w - 1 \quad (3)$$

where:

- X : is the horizontal extension in grid cell units
- L : is the horizontal dimension of the viewport
- W : is the vertical dimension of the viewport
- w : the number of rows of the extent (vertical)
- l : the number of columns of the extent (horizontal)

If X is non-integer the number of columns to be added should be equal to the next largest integer. If X is negative then this means that rows must be added to the extent to achieve proportional dimensions. A similar formula provides the magnitude of the extension along the vertical dimension.

Determination of Space along the Border of the Extent

Up to this point we have transformed the study area into a format which can be easily handled by a declarative language. The study

area also is at the largest scale that the display area permits, and, finally, the area in the configuration is balanced. The next step of the process consists of detecting the empty space surrounding the study area into which the remaining map components can be placed. That empty space is represented in a data structure such that:

- no useful empty space is undetected,
- the identification of size and shape of blocks of empty space is easily determined, and finally,
- translations and scale changes can be made easily using a declarative language.

The detection of useful empty space can be made if we scan the rectangle, which results from the adjusted extent in the four cardinal directions. For each row of a given scan direction, the run length is determined. The run length is defined as the number of empty cells from the frame to the first non-empty cell which is met in the direction of scanning. Scanning Figure 5, for example, from left to right would result in run lengths of: 3 for row 1, 3 for row 2, 3 for row 3, and zero for the remaining rows. Notice that a left-to-right scan does not detect all empty cells (e.g. 2,4). These cells, however, can be detected when the extent is scanned from another direction. Continuing with the previous example, and scanning from bottom to top, the corresponding run lengths are : 3 for column 1, 4 for column 2, 5 for column 3, and zero for all other columns. The empty space at the top and right sides of the extent are detected when Figure 5 is scanned from top to bottom and from right to left. The only empty space which is outside the border of the study area, and is not detected by any scan, is cell (6,4). Such empty space is unlikely to be useful for placing map components.

Since we have identified a way for detecting the location, size, and shape of useful empty space, the next problem that must be addressed is the efficient representation of that information for subsequent manipulations. This problem is solved by using a data structure to store the direction and run length of each swath along each scan direction. A swath is either a row or a column of the adjusted extent depending on the direction of the scan. An abstract data structure which enables the representation of all information related to a scan direction is shown in Figure 6. Specifically, empty space is represented as a compound object, which contains the scan origin, the scan destination, and a list of objects containing information about the swath number and its run length. Figure 6 also displays how the data structure can be implemented as a PROLOG fact. Tables 1.1, 1.2, 1.3, and 1.4 show the information stored in the database if our procedures are applied to the area shown in Figure 5. Each table is a separate fact, and these four facts suffice to store all information needed for evaluating the useful empty space of a configuration.

Geometrical Transformations using the Data Structure

In this section it will be demonstrated how placement of map components can be made using scale change and translation operations and the data structure shown in Figure 6.

Change in scale. In this case we deal only with scale reductions with respect to the frame of an area. Although enlargement can be

treated in a similar fashion we assume that we start with the largest possible scale of the study area and reduce it, until we achieve a satisfactory allocation of map components. The input of the scale change procedure are four facts describing the empty space around the border of the study area. The output of the procedure consists of four new facts describing the empty space at the new scale. Conceptually, reduction with respect to the frame can be made by adding an appropriate number of rows and columns around the adjusted extent. This increase in the frame size allows us to reduce the scale, and helps to avoid difficulties resulting from the raster format of the data.

To keep the dimensions of the frame proportional to the dimensions of the display, we determine the number of rows and columns added at each scale change. If L and W are the horizontal and vertical dimensions of the final display, l and w are the horizontal and vertical dimensions of the adjusted extent, and X and Y are the number of columns and rows that must be added in the frame to reduce scale, then the equation relating X to Y is:

$$X = (L / W) * Y = (l/w) * Y \quad (4)$$

To overcome problems resulting when X (or Y) is non-integer, it is suggested that Y be chosen first if $W \leq L$. If X is non-integer then the next largest integer is chosen. If, on the other hand, $W > L$ then the value of X should be chosen first. To keep the whole image balanced after the addition of new columns and rows, half of the rows and columns are distributed at the bottom and left sides and the other half at the top and right sides of the image. An illustration of this operation is shown in Figure 7.

After the addition of new rows and columns the empty space and thus the run length of each swath has changed. If XL , XR , YB , YT indicate the number of swaths added to the left, right, bottom, and top sides of the adjusted extent respectively then, for the case of left to right scan, the following rules determine the numbering and the run length of each swath. Rules for other direction scans are specified in a similar fashion.

Left to right scan:

- Add YB elements at the head and YT elements to the tail of the list describing the empty space of the swaths. The swath numbers of the new elements are integers and are selected such that the elements of the list are in ascending order with respect to the swath numbers. The run length of each new swath is set to be equal to $XL + l + XR$ where l is the horizontal length of the frame of the study area before the addition of new swaths.

- For every other element of the list :

- If the run length of that element is equal to l then set the run length equal to $XL+l+XR$
 - otherwise set the run length equal to $XL+RL$
 - where RL is the old value of run length for that swath.

If we choose $XL=1$, $XR=1$, $YT=1$, $YB=1$, $l=8$, $w=8$ and we apply the previous rules to Table 1.1, we will get Table 2.1. Applying similar rules corresponding to the other three scan directions to Tables 1.2, 1.3, and 1.4, we obtain Tables 2.2, 2.3, and 2.4. Notice that the last tables describe the useful empty space of the picture displayed in Figure 7.

Translation to the right. Conceptually, translation of the study area to the right with respect of its frame can be accomplished by adding a number of columns at the left and deleting an equal number of columns from the right side of the current frame (see Figures 7 and 8). This operation simply changes the relative position of the objects with respect to the frame. If XL is the number of columns which are added to the left side of the frame and the other variable names have the same interpretation as in 3.6.1, then the rules determining values for the new empty space are:

- a Left to right scan.
 - For each element of the swath list
 - If the run length of the swath is less than l
 - then the new run length is $RL + XL$
 - otherwise the run length remains the same.
- b Top to bottom scan.
 - Add XL elements with appropriate swath numbers at the head of the list containing information for the run length of each swath. The run length of each new element is w.
 - Delete XL elements from the end of the list. Only elements which have run length equal to w are allowed to be deleted. Elements with run length less than w can not be deleted because the corresponding swath contains shaded cells and thus it is part of the boundary of the study area.

Rules applied to the right to left and bottom to top scan are analogous to a and b respectively. Applying such rules to Tables 2.1, 2.2, 2.3, 2.4 and taking $XL=1$ and $w=10$, Tables 3.1, 3.2, 3.3, 3.4 are derived. Those tables represent the useful empty space of Figure 8.

Empty Block Size Determination

Information stored in the data structures representing empty space can be manipulated to determine if a map component fits there. If, for example, a 3 by 3 legend must be placed in Figure 5, the solution is to put that legend at the bottom left. That block of empty space can be identified if the values of Tables 1.1 through 1.4 are examined. If the algorithm searches the swaths of Table 1.1 it will identify that there are three consecutive swaths which have run length equal to three -swaths 1, 2, and 3. It then can infer that there is a 3 by 3 square of suitable empty space at the left side of those swaths. If, on the other hand, the dimensions of the legend were 3 by 4, the algorithm will fail to find a solution by traversing Tables 1.1 through 1.4. Since no translation can be performed in Figure 5, the next step is to do a scale reduction. Figure 7 shows the result of such a reduction, and Tables 2.1 through 2.4 represent the empty space of that figure. Searching the swaths in Table 1.1, the algorithm can now identify four consecutive swaths with run length greater than or equal to 3 -swaths 0, 1, 2, 3- and it will come up with an answer. Similar reasoning can be applied for cases where more than one map component must be allocated. In addition to performing list searches, translations, and scale changes, the program keeps track of space reserved for components that have already been allocated.

CONCLUSIONS

In this paper the problem of allocating map components within a viewport has been explored. A three stage approach is adopted to determine the final layout for a simplified study area. The preprocessing stage was the principal topic of our discussion. Data representations and procedures, which allow satisfactory allocation of map components using rules, were described. The effectiveness of the chosen representations is illustrated using examples. The second stage of the approach, which consists of the determination of rules guiding the placement of map components, is currently a focus of our work.

REFERENCES

- Broome, F.R., Beard, C., and Martinez, A.A. 1987. Automated map inset determination. Proceedings, Auto-Carto 8, pp. 466-470.
- Cuff, D.J., and Mattson, M.T. 1982. Thematic Maps: Their Design and Production. New York: Methuen.
- Dent, B.D. 1985. Principles of Thematic Map Design. Reading: Addison-Wesley.
- Fisher, P.F., and Mackaness, W.A. 1987. Are cartographic expert systems possible? Proceedings, Auto-Carto 8, pp. 530-534.
- Foley, J.D., and Van Dam, A. 1982. Fundamentals of Interactive Computer Graphics. Reading, MA: Addison-Wesley.
- Mackaness, W.A. and Fisher, P.F. 1987. Automatic recognition and resolution of spatial conflicts in cartographic symbolisation. Proceedings, Auto-Carto 8, pp. 709-718.
- Maggio, R.C. 1987. The role of the geographic information systems in the expert system. Proceedings, GIS '87, pp. 685-692.
- Monmonier, M.S. 1982. Computer-Assisted Cartography: Principles and Prospects. Englewood Cliffs: Prentice-Hall.
- Morse, B.W. 1987. Expert system interface to a geographic information system. Proceedings, Auto-Carto 8, pp. 535-541.
- Robinson, A.H., Sale, R.D., Morrison, J.L., and Muehrcke, P.C. 1984. Elements of Cartography (5th ed.). New York: John Wiley.
- Robinson, G. and Jackson, M. 1985. Expert systems in map design. Proceedings, Auto-Carto 7, pp. 430-439.

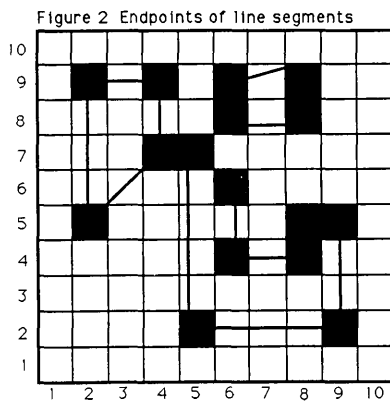
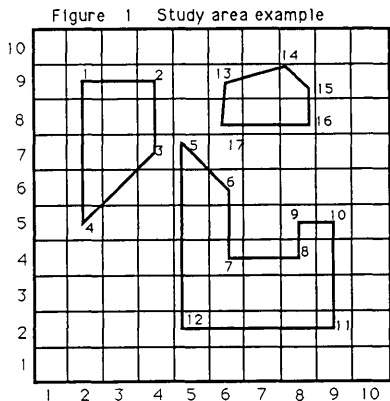


Figure 3 Structure for representation of cells

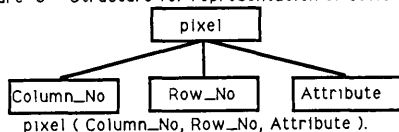


Figure 4 Border of the study area

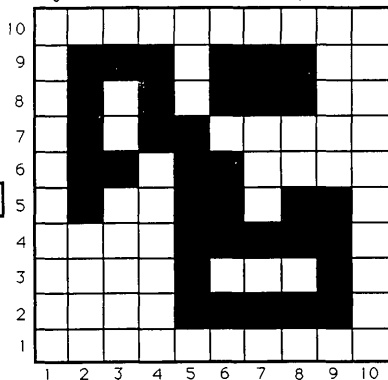


Figure 5 Extent of the area

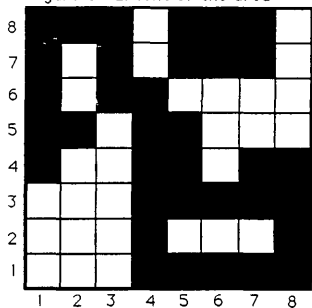


Table 1.1

Scan origin		left							
Scan destination		right							
Swath_no	th	1	2	3	4	5	6	7	8
	Run_length	3	3	3	0	0	0	0	0

Table 1.2

Scan origin		top							
Scan destination		bottom							
Swath_no	th	1	2	3	4	5	6	7	8
	Run_length	0	0	0	2	0	0	0	4

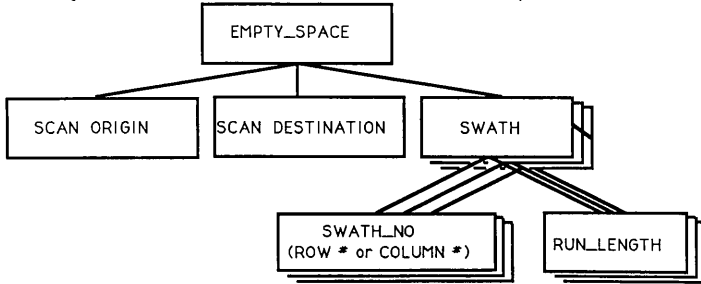
Table 1.3

Scan origin		right							
Scan destination		left							
Swath_no	th	1	2	3	4	5	6	7	8
	Run_length	0	0	0	0	3	4	1	1

Table 1.4

Scan origin		bottom							
Scan destination		top							
Swath_no	th	1	2	3	4	5	6	7	8
	Run_length	3	4	5	0	0	0	0	0

Figure 6 Data structure for representation of empty space



empty_space(Scan_origin,Scan_destination,[swath(Swath_No, Run_Length)])

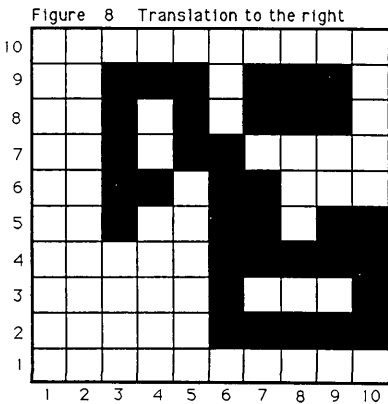
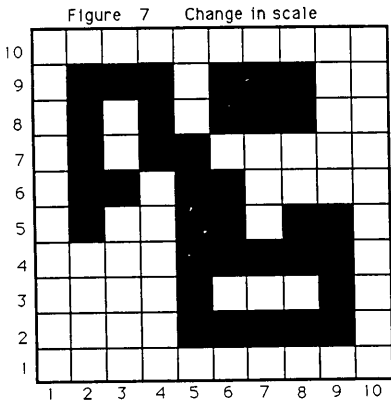


Table 2 1

Scan origin	left
Scan destination	right
Swath_no	0 1 2 3 4 5 6 7 8 9
Run_length	10 4 4 4 1 1 1 1 1 10

Table 2 2

Scan origin	top
Scan destination	bottom
Swath_no	0 1 2 3 4 5 6 7 8 9
Run_length	10 1 1 1 3 1 1 1 5 10

Table 2 3

Scan origin	right
Scan destination	left
Swath_no	0 1 2 3 4 5 6 7 8 9
Run_length	10 1 1 1 1 4 5 2 2 10

Table 2 4

Scan origin	bottom
Scan destination	top
Swath_no	0 1 2 3 4 5 6 7 8 9
Run_length	10 4 5 6 1 1 1 1 1 10

Table 3 1

Scan origin	left
Scan destination	right
Swath_no	0 1 2 3 4 5 6 7 8 9
Run_length	10 5 5 5 2 2 2 2 2 10

Table 3 2

Scan origin	top
Scan destination	bottom
Swath_no	-1 0 1 2 3 4 5 6 7 8
Run_length	10 10 1 1 1 3 1 1 1 5

Table 3 3

Scan origin	right
Scan destination	left
Swath_no	0 1 2 3 4 5 6 7 8 9
Run_length	10 0 0 0 0 3 4 1 1 10

Table 3 4

Scan origin	bottom
Scan destination	top
Swath_no	-1 0 1 2 3 4 5 6 7 8
Run_length	10 10 4 5 6 1 1 1 1 1

AUTOMATED INSETTING:
AN EXPERT COMPONENT EMBEDDED IN
THE CENSUS BUREAU'S MAP PRODUCTION SYSTEM

April A. Martinez
Geography Division
U. S. Bureau of the Census
Washington, DC 20233

ABSTRACT

The U.S. Bureau of the Census operates in a mass map-production environment, producing hundreds of thousands of unique, individual maps in batch mode. Given a very short window of time, interactive cartographic decision-making is impossible. Consequently, the Census Bureau has developed, and continues to enhance, "expert" components for its automated map production system. These expert components represent the quantification of collective cartographic decision-making processes. In use at this time are automated names placement, automated scaling, and automated inseting.

This paper discusses the development of one of the expert components: automated inseting. It describes how cartographers at the Census Bureau identify and quantify approaches for inset requirements, and examines how they developed the automated system once the approaches were identified and rules were established. The paper also addresses plans for future improvements of the existing system.

INTRODUCTION

Most cartographic design decisions are made based upon scientific and artistic rules of cartography. Cartographic design is a process of selection of scale, symbolization, and so forth. (Dent, 1985) The rules are continually applied while producing a map in an interactive automated environment. When, however, many thousands of maps are produced by automated means and constrained by limited time schedules, cartographic interaction is impossible and the work must be done in batch mode. It then becomes necessary to design an automated system in such a way as to imitate a cartographer's decisions, thereby developing an expert system.

The expert cartographic system should reflect the cartographer's decision-making process. The most important and difficult component of developing an expert system is the quantitative definition of the cartographer's reasoning process, particularly the implementation of aesthetic preferences.

At the Bureau of the Census, production of the maps for the 1990 census data collection is totally automated. Interactive cartographic decision-making is impossible. There are at least eight different types of maps, each having nationwide coverage, resulting in an estimated

2.5 million unique map sheets. Production of these maps began in April 1988 and will be completed by March 1991. This tight time frame prohibits cartographers from interactively designing the maps. To produce the maps in batch mode, the Bureau of the Census has developed "expert" components for its automated mapping system.

Background

One of the "expert" components developed for the automated map production system was insetting. It grew directly from the Census Bureau's first attempts to automatically determine an adequate map scale. The traditional scale determination rule for one type of field map was that the length of the smallest side of census blocks must be at least one inch to accommodate "map spotting."* The direct application of this rule in production consistently resulted in map scales too large for most of the geographic areas mapped, consequently yielding many more map sheets than were required, and than a manual system would have produced.

To reduce the number of map sheets, the Census Bureau decided to change the scale determination rule by forcing only 60 percent of census block sides to equal at least an inch. This change produced numerous isolated areas on the map for which the scale was too small and unsuitable for map spotting. Adequate map scale to allow enumerators to successfully complete their field assignments was required. A scale too small for specific areas on a map limits the use of the map. As a result automated insetting was developed.

In the present environment, the production system first determines the latitude and longitude limits of the geographic entity being mapped, followed by the scale of the map. Map scales are not standard. They are determined by the use of the map and the feature content of the geographic area within the window. (Martinez, 1987) The scale chosen is considered the best scale for the majority of the area within the window. This means there can be areas within the window for which the determined scale is too large or too small. Areas within the window for which the map scale is too small are selectively plotted on separate sheets at larger scales. These areas are selected by the map production system's insetting component.

DEVELOPMENT

Automated insetting attempts to emulate the steps a cartographer takes when deciding which geographic areas within a map to inset. Defining the decision-making process was the first phase of development. After reviewing numerous maps and selecting areas that required insetting, cartographers at the Census Bureau found that

* A map spot is a dot penciled with an associated number on an enumerator map by a census enumerator to show the location of a housing unit, multiunit structure, special place, or business establishment for field operations.

their decision to inset is based upon the answers to three questions:

1. What constitutes dense areas?
2. Which features are to be considered when determining density?
3. What determines the extent of the insetted area?

Density

On 1990 census maps used for field operations, the driving factor for insetting is "feature density." Because of the nature of field operations, most of the features in the Census Bureau's Topologically Integrated Geographic Encoding and Referencing (TIGER) File must be shown on the maps. The maps are used for a variety of census operations. Some are used to depict political or statistical area boundaries, such as a county or census tract, while others are used for map spotting. One map type in particular must label and display every census block within a geographic entity. Because of map requirements cartographic features on these maps cannot be generalized.* Therefore, the major problem for an automated insetting system is defining the density of features on the map and insetting those areas whose feature density adversely affects the use of the map for a specific operation. To meet this requirement, the original programs written for the insetting system attempted to allow a computer to analyze the density of the features on a map similar to the way a cartographer views the map.

Numerous factors were considered in defining areas that are considered to be "too dense." These were narrowed to eight factors quantifiable from data in the TIGER File. Eight algorithms for depicting density were developed and tested. The TIGER File structure is based upon entities known as 0-cells, 1-cells, and 2-cells. For a full description of these entities see referenced papers.** For the purpose of this paper, 0-cells can be considered as endpoints of a 1-cell, which in many cases are intersection points of line features. The 1-cells are lines connecting 0-cells, and can serve as bounding segments of a 2-cell. Two-cells are the smallest areas bounded by 1-cells. Aggregates of 2-cells make up higher-level census geography, beginning with census blocks. (Beard, Broome, and Martinez, 1986, 2)

The algorithms were:

1. One-cell, midpoint method. The coordinates of the endpoints of the 1-cells are added together and divided by two to get a midpoint.
2. One-cell, average of all points method. The coordinates of all the points along the 1-cell and the endpoints are added and the results divided by the count to get an average.

* "Generalized" in this sense refers to selectively eliminating specific types of cartographic features from a map to reduce clutter.

** See referenced papers by Broome, Kinnear, Boudriault, and McDowell for discussions on the TIGER System.

3. One-cell, endpoint method. The 0-cells are used.
4. Two-cell, envelope midpoint method. The maximum and minimum coordinates of the 2-cells are added and divided by two to get a midpoint.
5. Two-cell, weighted area centroid method. The area and geographic centroid of the 2-cell is determined and the centroid is assigned the value of the area.
6. Census block envelope midpoint method. The maximum and minimum coordinates for each census block are determined by aggregating the 2-cells that constitute the block. The sum is divided by two to determine the coordinates.
7. Census block 2-cell average centroid method. The 2-cell average centroid is derived by adding all the maximum and minimum coordinates of the 2-cells and dividing by the count to get an average centroid.
8. Census block weighted area centroid method. The area and geographic centroid of the block is determined and the centroid is assigned the value of the area.

A program was written to test each method. Production statistics such as processing time and computer cost were recorded for each algorithm. (Beard, Broome, and Martinez, 1986, 3)

After plotting the points from the intermediate files, four of the methods were retained for further research, because they were computationally the most efficient and observationally determined to best identify dense areas of features.* Of the eight methods, the four retained for determining feature density were:

1. The 1-cell midpoint;
2. the 1-cell average of all points;
3. the 2-cell, envelope midpoint; and
4. the block envelope midpoint method.

Further tests were made and analyzed for the ability to determine dense areas. The algorithm developed to examine density is based upon count of calculated points within a grid cell of a predetermined size. The size of the grid cell is related to the specific use of the map. For example, the size of the grid cell for a map that needs every polygon labeled with a block number that is approximately 0.24" wide and 0.08" tall, is 0.25" X 0.25"; this is smaller than the grid cell for a map that requires all linear features to be at least an inch long.

*Insetting based solely on the size of an area being labeled is an alternative method of insetting.

The count of occurrences within the grid cells for each method was stored in a large matrix. The resulting matrix of values was then smoothed by summing the counts of groups of nine grid cells and recording the total as the value for the center cell of the 3 x 3 group. This smoothing operation removes local irregularities due to the use of a single coordinate to represent a linear and/or areal feature (see figure 1). The matrix of smoothed grid values for each map was plotted and visually analyzed. By classing the grid values and shading the classes on the matrix, it became evident that every file clearly represented the general feature pattern of the map.

The next major step in developing the automated inseting system was to introduce cartographic expertise. This provided a way to interpret the grid cell values. Ten professional cartographers, with an average experience of five years in census map design, examined several plotted maps. They agreed that the need for an inset was related to the use of the map. Knowing the use of the maps, they were asked to mark on overlays the geographic extent of areas that required insets. In order to have a valid sample cartographers worked independently and did not discuss their results until after the test. In addition, none of the cartographers were shown the matrix of grid cell values created for each map.

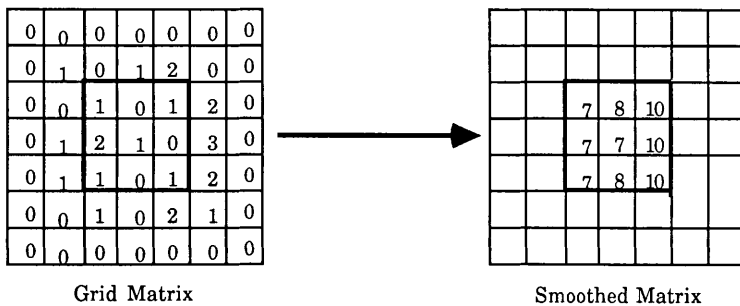


Figure 1. This is a portion of a map grid matrix. It shows the count of 1-cell midpoints that fall within the 0.25" grid cell. Summing the count of 3 x 3 grid cells and recording this total as the value of the center cell creates a smoothed matrix.

A comparison of the overlays revealed a close match between the choice made by the cartographers. The variances averaged less than one-fourth inch at map scale. This variance was considered acceptable for the intended purpose. So the inset overlays that the cartographers produced were registered to each other, and "average" inset overlays were produced.

Rules for density

The "average" inset overlays were registered to the plotted matrices of grid values of the same maps to see if any relationship could be established. A visual examination revealed that each of the four matrices appeared to have a specific breakpoint value; that is, a value

within a grid cell that, if surpassed, was deemed too dense on the cartographers' average inset overlays. This density (breakpoint) value varies for each map type by:

1. The method used to determine density;
2. the grid cell size; and
3. the purpose of the map.

With known breakpoint values, the major obstacle of automated inseting was overcome; the system was able to determine feature density.

In production, the system creates, then scans, a grid cell matrix for cells with values higher than the breakpoint value. If higher values are found, the system checks systematically around the cell for adjacent cells that have values higher than that defined for an operation. If an adjacent cell is found, the system moves to this grid cell position and again checks for an adjacent cell, continuously counting the number of cells and storing the maximum and minimum grid cell position values. Finally, the system returns to the starting cell position. The system stores the results as a possible inset window only if it finds at least three cells adjacent to each other.

Rules for qualify

A map can have numerous insets based on feature density. However, some of these insets may be unnecessary depending on the census operation using the map. A major concern for field maps is the number of map sheets; if there is any way to reduce the number of sheets without affecting the quality of the census operation or the usability of the map, then an attempt is made to reduce the number of sheets. One way to reduce the number of sheets is by eliminating insets. Therefore, before a possible inset window determined by density alone can become an inset, it must qualify based on rules of inset qualification.

The rules of inset qualification were and are being determined through discussions with cartographers pertaining to the map information on which they base their judgement for eliminating insets. Some of the rules defined are general and are applied to every map generated in production. For example, some maps show features in fringe area outside the geographic area being mapped; however, it is unnecessary to have insets of dense areas within the fringe, so one of the first rules of qualification is that an inset must be within the geographic area being mapped (see figure 2).

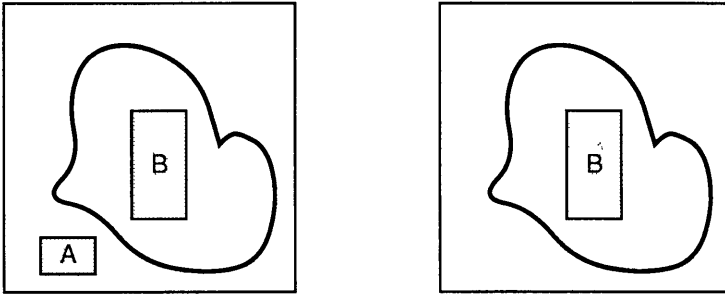


Figure 2. Example of an inset qualification rule: Only keep insets within the entity being mapped. Inset "A" is removed because it is outside the entity boundary.

Other rules for inset qualification are specific to the map use. A rule for the map that labels every polygon with a block number is that a possible inset window qualifies as an actual inset only if at least five whole census blocks are within it. (It was reasoned that block numbers can be arrowed from adjacent blocks into clusters of fewer than five blocks without significantly hindering readability.) In another operation, the map is used by officials to verify and update the location of their government's political boundaries; consequently, for that map type only insets with political boundaries would qualify as actual insets.

The rules of qualification are the most powerful tool within the automated inseting system. By defining the rules of inset qualification based upon anticipated map use, the cartographer affects the entire design of the map and the effectiveness of the census operation for which the map is intended.

Rule for nearness

A cartographer manually creating a map visually determines which areas on a map are too dense at the desired map scale. Some of the areas are close enough to other potential areas to be grouped together in one inset. The cartographer also visually determines the best geographic extent of an inset. The insetted area can be either an amorphous shape or a rectilinear window. The cartographer knows where to stop expanding the limits of the inset in either case.

In the automated system, the rule of nearness was defined by again having cartographers review numerous maps for which they determined a specific distance at map scale that constituted nearness on a map. The distance was further redefined during test production when it was decided that the system was combining too many insets, thereby creating large, multisheet insets.

The limits -- or the latitude and longitude windows -- of the actual insets were the maximum and minimum limits of the grid cells making up each inset. However, it was found that an inset's limits many times were coincident with part of an incorporated place. As a result, the inset limits were expanded by one-fourth of the nearness

distance in an attempt to inset the entire incorporated place. This results in a geographic area portrayed entirely at the larger inset scale.

In production, once the system determines if all the inset windows qualified based on the inset qualification rules, it eliminates insets that fall within another inset. The system also expands windows of insets that overlap each other. The result is one larger inset window (see figure 3). Finally, the system checks if multiple insets are close enough to each other to merge to become one larger inset.

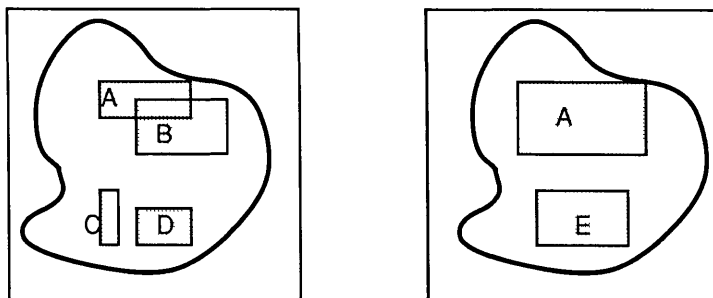


Figure 3. Example of overlapping insets and insets near each other merged to make larger insets.

ENHANCEMENTS

An automated cartographic mapping program used in batch production needs an intelligent insetting system. Insetting by feature density is the most common method used at the Bureau of the Census. The rules are defined by map use, which varies by map type. The major rules for insetting are first to define the density breakpoint value for the map based on grid cell size and algorithm used to determine feature density. Second the inset must qualify based on the rules of inset qualification. Finally, the extent of the inset must be determined.

Modifications to the rules and algorithms are applied as new and unanticipated circumstances arise. New density values for census operations have been defined and rules have been added to inset qualification. Most enhancements to the system are in the rules for inset qualification.

At this time, the Bureau of the Census is beginning to develop a publication mapping system. One area of research involves the feasibility of having the automated insetting system inset entire incorporated places if any portion of an incorporated place falls within an inset. This could eliminate the need to create new plot files solely for the purpose of creating maps for each incorporated place; the inset would serve as a larger-scale inset area within a county map as well as an incorporated place map. However, many issues need to be

addressed on the overall affect of this concept. New cartographic rules would be required for the system. For example:

How does the system handle urban sprawl? Should the system inset a network of features that is not dense simply because it is within the limits of an incorporated place?; if not, where does one stop expanding the inset? How does one handle corporate corridors? How is the number of sheets affected?

What happens if more than one place falls within an inset window? Does the system make two insets, or does it put both incorporated places on one inset map? Should it make an inset of the incorporated place that covers the biggest portion of the original inset?

Will the whole map become an inset?

These are only a few of the questions that are being considered, as additional rules for the system are being defined and algorithms developed.

CONCLUSION

Map production for the 1990 census maps began in spring of 1988. Automated inseting has been successfully implemented for every operation to date. In batch production the insetting component first determines possible insets by density. Once these have been selected, the system then checks if the inset qualifies, and finally it merges insets that are within or near each other.

The system has grown rapidly from its original intent; however, the basic concept of a system defining feature density and insetting areas with areas of high feature density is still the same. As the insetting system continues to be used, new rules are defined and added to the component cartographically enhancing the quality of the insets defined by the system. Future developments of automated mapping systems for batch production must include inset determination if for no other reason then to tell the system when a given map has areas too dense for effective use.

REFERENCES

- Beard, C., Broome, F., Martinez, A. 1987, Automated Map Inset Determination: Proceedings Auto Carto 8, pp. 466 - 470.
- Boudriault, G. 1987, Topology in the Tiger File: Proceedings Auto Carto 8, pp. 258 - 263.
- Broome, F. 1984, Tiger Preliminary Design and Structure Overview: The Core of the Geographic Support System for 1990: presented at the 1984 Annual Meeting of Association of American Geographers.
- Dent, B. 1985, Principles of Thematic Map Design: Addison-Wesley, p. 22.

Hammond, R., McCullagh, P. 1978, Quantitative Techniques in Geography: An Introduction: Oxford University Press, pp. 36 - 40.

Kinnear, C. 1987, The TIGER Structure: Proceedings, Auto Carto 8, pp. 249 - 257.

McDowell, T., Meixler, D., Rosenson, P., Davis, B., 1987, Maintenance of Geographic Structure Files at the Bureau of the Census: Proceedings Auto Carto 8, pp. 264 - 269.

Martinez, A. 1987, Applications of Expert Rules in Automated Cartography", presented at the 1987 Applied Geography Conference, pp. 249 - 257.

ACCESSING SPATIOTEMPORAL DATA IN A TEMPORAL GIS

Gail Langran
Department of Geography, DP-10
University of Washington
Seattle, WA 98195
bitnet: langran@uwav1

ABSTRACT

This paper evaluates ways to boost the performance of spatiotemporal data access in a temporal GIS via strategic partitioning and indexing. Using a specific conceptual model as an example, the discussion describes a taxonomy of access methods and explores the methodological options available.

INTRODUCTION

Effective access methods for spatiotemporal data are vital to the development of effective temporal geographic information systems. Because the access scheme of an atemporal GIS has major performance impacts, we can expect the spatiotemporal corollary to be similarly influential upon temporal GIS performance.

Although many alternate methods of representing temporal geographic data surely exist, this discussion builds upon a specific conceptual model that is described elsewhere in some detail (Langran and Chrisman 1988), and which was first suggested by Chrisman (1983). This model, called a space-time composite, represents spatiotemporality by accumulating geometric change into one integrated topological description. Each successive change causes the changed objects to break from parent objects, creating new objects with histories distinct from those of their neighbors. In other words, the representation decomposes over time into the area's greatest common spatiotemporal units; each unit's history is described by a variable-length list of attribute sets bracketed by effective dates.

Figure 1 shows a space-time composite of temporal census-tract data. Polygons 1, 2, 3, and 4 are the greatest common spatiotemporal units of Census Tracts A, B, and C. Until 1980, only three polygons represented the three census tracts because Polygon 3, being part of Tract B, was incorporated into Polygon 2. Then in 1980, the area represented by Polygon 3 moved from Tract B to Tract C, creating a fourth polygon with a distinct history from its neighbors.

We can reconstruct any time slice from the space-time composite by referencing the attribute histories of its objects to find the attribute sets that were current on the requested date. This tack obviously works for small numbers of objects with shallow histories, but it is reasonable to ask whether the data processing burden is manageable given realistic geographic data volumes. Clearly, a complex query could swamp a system without an access scheme to boost performance.

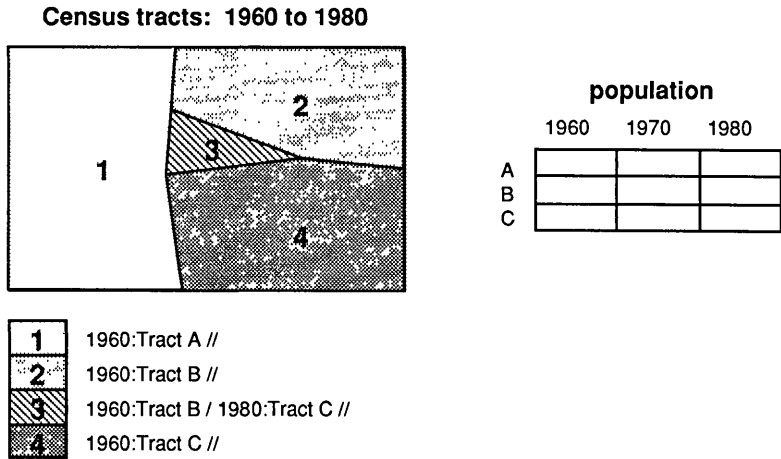


Figure 1. Temporal census-tract data represented as a space-time composite. Polygons 1, 2, 3, and 4 are the greatest common spatiotemporal units. Polygon 3 was part of Tract B until 1980, when it moved to Tract C. Census statistics such as population require no special treatment because time is symmetric with respect to other attributes.

QUERIES TO A TEMPORAL GIS

Table 1 lists potential queries to a temporal GIS. The listing distinguishes between forays into the past and future because the mechanisms for "examine" and "extrapolate" could differ depending on whether data are stored or computed.

Table 1. Temporal GIS queries.

- Examine an object's history.
- Extrapolate an object's future.
- Examine a single time slice.
- Examine an object's history; when the object meets some criteria, examine that time slice.
- Extrapolate an object's future; when the object meets some criteria, examine that time slice.
- Examine a single time slice; examine the histories of objects meeting some criteria.
- Examine a single time slice; extrapolate the histories of objects meeting some criteria.
- Examine the histories of all objects.
- Extrapolate the futures of all objects.
- Examine time slices, going backward through time.
- Extrapolate time slices, going forward through time.

If we ignore extrapolated information and concentrate on accessing stored data, four primitive queries lie at the root of Table 1's eleven queries:

- simple temporal query*, i.e., what is the state of an object at time t?
- temporal range query*, i.e., what happens to an object over a given period?
- simple spatiotemporal query*, i.e., what is the state of a region at time t?
- spatiotemporal range query*, i.e., what happens to a region over a given period?

Access Mechanisms for Query Response

The space-time composite describes all temporality via time stamps in the attribute database, which permits us to treat time aspatially and space atemporally. Changes to geometric objects spawn new objects and break existing objects; these fragments replace the previous unbroken versions. By definition, each object in the space-time composite has a single geometric and topological description throughout time.

Thus, the temporal access mechanism for a space-time composite operates primarily on an attribute database, which is cross-referenced to the spatial representation. For the sake of simplicity, this discussion assumes that the attribute database is relational (here called an RDB) and that tuple versions correspond to object versions. Current RDBs can be fortified by a suite of indexing methods which, unfortunately, are almost solely one-dimensional (Freeston 1987). By stepping through the algorithms required to respond to the four primitive queries, we can comprehend how their requirements exceed standard RDB and GIS accessing capabilities.

Simple Temporal Query. The goal of the simple temporal query is to find an object version that was current on a specified date. Ideally, the system could search for the tuple whose "Object = ID" and "Time = T." But a temporal database is event-driven so a tuple will not necessarily have a time value to match every T. In essence, an object lifespan can be considered a chain whose nodes are the object's birth and death, and whose vertices are the points where the object changed. To time slice the object is to locate the value of a point along that chain based on the time stamp that equals or *immediately precedes* the requested time. This implies that ordering an object's versions within storage would be helpful.

Temporal Range Query. To respond to a temporal range query, the system must locate all versions of the desired object that were current during any part of the specified time span, i.e., where Object = ID, Time < Maximum Time, and Time > Minimum Time. The system could select all qualifying tuples; or if tuples are temporally ordered, the system could locate a tuple at one end of the desired time range then "walk" through time-sorted object versions until reaching the other end of the range.

Spatiotemporal Queries. The preceding two queries focus on single objects that meet temporal criteria. In contrast, spatiotemporal queries request all objects within specified spatial and temporal ranges. To respond to a simple spatiotemporal query, the system clips the desired region from the space-time composite, locates all attribute records for the desired region as of the desired time, then dissolves the chains that separate polygons of like attributes (i.e., recomposes the greatest common spatiotemporal units into greatest common spatial units). Alternately, the system can first access the required attribute records, then match them to the space-time composite. Responding to a spatiotemporal range query is the same as a simple spatiotemporal query except the system seeks attribute records falling within a space-time range.

The Search Space of the Four Queries

While the data space of these four queries is three-dimensional (two space and one time dimension), a closer look reveals that the queries define ranges of zero, one, two, and three dimensions, respectively. Query One defines a degenerate range, the point in data space that describes an object's state at a given time (for example, the location of a tuple that references the attributes current at that time). The data that satisfy Query Two lie along a vector that traces the changes undergone by a specific object over a specific period--a one-dimensional range. The response to Query Three occupies an orthogonal plane--a two-dimensional range. And the objectives of Query Four are located in a cube embedded within the GIS data space defined by a one-dimensional time range and a two-dimensional space range (Figure 2).

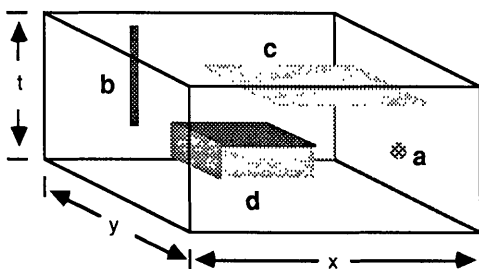


Figure 2. The search space of four primitive geographic queries in a three-dimensional data space. (a) A simple temporal query references a single point in space. (b) A temporal range query references a vector along which the desired data lie. (c) A simple spatiotemporal query references a plane within which the desired data lie. (d) A spatiotemporal range query references a cube within which the desired data lie.

RESPONDING TO TEMPORAL GIS QUERIES

Understanding the number of dimensions involved in a query is helpful in selecting the most effective means of response. Noronha (1988) and Chrisman (1988) provide recent discussions of multidimensional data access methods. Unfortunately, many methods that are termed "multi-" or "k-dimensional" can access only zero-dimensional ranges in a multidimensional data space. While such methods hardly seem worthy of their name to those accustomed to working with multidimensional ranges in multidimensional space, their intent is to facilitate "composite key" or "multikey" retrievals. Such retrievals seek a subset of records in k-dimensional space, one dimension per attribute. Table 2, drawing on Noronha's discussion, lists methods designed to access data in a multidimensional data space according to the number of ranges each can treat.

Given the accessing options available for zero-, one-, and higher-dimensional ranges, the next step is to examine the methods that treat k-dimensional ranges in k-dimensional space (Table 2's third column), since these are the methods that potentially provide a single means of rapid response for the four primitive queries of a temporal GIS.

Table 2. Accessing ranges of zero, one, and more dimensions in k-dimensional space.

Zero-dimensional	One-dimensional	K-dimensional
K-d tree ¹	Strip tree ¹⁰	R-tree ¹¹
K-d-b tree ²		R+ tree ¹²
Multikey hashing ³		Packed R-tree ¹³
Extendible hashing ⁴		Cell tree ¹⁴
Point quadtree ⁵		Grid file ¹⁵
Multidimensional trie ⁶		BANG file ¹⁶
Multidimensional directory ⁷		BSP tree ¹⁷
Log log n structure ⁸		Region quadtree ¹⁸
Quintary tree ⁹		EXCEL ¹⁹
		Field tree ²⁰
		Quad-CIF tree ²¹
<hr style="border-top: 1px dashed black;"/>		
¹ Bentley 1975	⁹ Lee and Wong 1980	¹⁵ Nievergelt et al. 1984
² Robinson 1981	¹⁰ Ballard	¹⁶ Freeston 1987
³ Rothnie & Lozano 1974	¹¹ Guttman 1984	¹⁷ Samet 1984
⁴ Fagin et al. 1979	¹² Roussopoulos & Leifker 1985	¹⁸ Fuchs et al. 1980
⁵ Finkel & Bentley 1974	¹³ Faloutsos et al. 1987	¹⁹ Tamminen 1981
⁶ Orenstein 1982	¹⁴ Gunther 1986	²⁰ Frank 1983
⁷ Liou & Yao 1977		²¹ Kedem 1982
⁸ Fries et al. 1987		

A Taxonomy of Access Methods

A taxonomy of access methods would be useful to truly understand the options available. Several writers have attempted to classify data access schemes. Nievergelt et al. (1984) define two broad classes: a scheme can organize the data themselves or partition the embedding space. For example, the actual locations of objects in data space determine the branching of k-d trees and R-trees. Conversely, a grid file or quadtree subdivides when a predetermined sector of data space exceeds a predetermined maximum capacity. The Nievergelt framework is quite useful conceptually but enough hybrid schemes exist to make it something less than a taxonomy. Specifically, the quad-CIF tree associates the minimum bounding rectangles of its objects with cells of a recursively subdivided embedding space (Noronha 1988). Other schemes that resist the Nievergelt framework are the BANG file, cell tree, and extendible hashing.

Freeston (1987) defines a pragmatic classification of access schemes: tree structures, extendible hashing, and grid files. But where the Nievergelt framework is perhaps too conceptual, this classification is too technical. The strengths and weaknesses of quadtrees and grid files, and those of R-trees and BANG files, are more similar than those of quadtrees and R-trees, or grid files and BANG files. Yet Freeston's classification results in the latter two dissimilar groupings based on common data structuring mechanics.

Noronha, too, defines a classification scheme for access methods (1988). This scheme distinguishes hierarchical vs. nonhierarchical and regular vs. object-oriented subclasses to highlight the major performance differences among methods. However, Noronha's goal is description and exposition, and he purposely avoids the rigor of a taxonomy.

The ideal taxonomy should produce clear distinctions between classes, and members of a taxonomic class should share strengths and weaknesses. The terminology used here departs from earlier conflicting, and potentially confusing, usages. The "access methods" described here have been variously termed "partitioning" (by Noronha), "indexing" (by Chrisman) and "file structuring" (by the bulk of computer scientists). This taxonomy distinguishes indexing from partitioning because these two operations are associated with separable functions and ramifications. Membership in more than one indexing class is permitted, but the partitioning class is uniquely defined, since herein lies the greatest performance distinction.

Indexing. The two major indexing methods are search trees and hashing. A search tree stores physical locations of entities according to some order; hashing methods use functions to compute storage locations. Topological navigation is a third indexing method that is somewhat peculiar to geographic applications. If data records have an innate order and supply pointers to neighbors, accesses within neighborhoods can use these "topological" data as stepping-stones to navigate from one object to another. The DIME file editor (White 1974), the ETAK automobile navigation system (White 1987), and the TIGRIS editor (Herring 1987) demonstrate topological navigation in a spatial system. Lum et al. (1984) demonstrate navigation in a temporal (and aspatial) RDB.

Partitioning. The most complex portion of the taxonomy describes partitioning. The wide array of partitioning strategies and problems are discussed at some length in both Chrisman 1988 and Noronha 1988. Partitions can be at one level or hierarchical. Each can use regular or irregular units. Irregular units can intersect or not within a level. Successive levels of a regular hierarchy might nest fully or not. Figure 3 depicts the relationship of these subclasses.

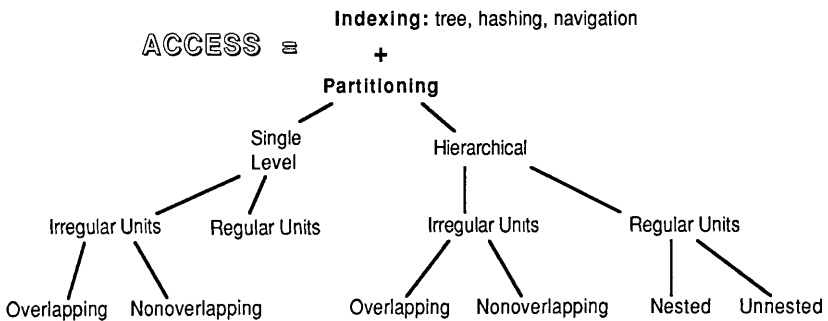


Figure 3. A taxonomy of multidimensional access schemes, which provides separate indicators for indexing and partitioning method.

Using the Taxonomy

The usefulness of the taxonomy is in evaluating a given access scheme for a given purpose. Since an access scheme inherits the strengths and weaknesses of its methodological components (as defined by the taxonomy), we can quickly judge a scheme by knowing its classification.

For example, irregular subdivisions permit us to cluster data as desired and thereby avoid splitting individual objects. Objects that are split by regular cells can be difficult to reconstitute and may produce erroneous replies to analytical queries concerning size or duration. Conversely, irregular cells require more complex heuristics to build and greater storage overhead to describe than do regular cells.

CONCLUSIONS

Four primitive queries whose ranges are zero-, one-, two-, and three-dimensional lie at the root of more sophisticated requests for temporal GIS information. A set of multidimensional data access schemes exist that are theoretically capable of boosting temporal GIS performance. These queries and access schemes are two endpoints from which to proceed to untangle the temporal GIS problem that lies between them.

This paper presents a taxonomy that groups access methods according to performance traits. Each subclass of the taxonomy is associated with a set of strengths and weaknesses, which are inherited by its members. The question, then, remains: what strengths would be most useful to a temporal GIS application, and what weaknesses would be intolerable? The cursory analysis described here indicates that using a topological navigator to supplement a tree or hashed index would assist in time-slicing temporal GIS data. How to partition spatiotemporal data is far more problematic and requires further examination.

ACKNOWLEDGMENTS

I would like to thank Intergraph Corporation's Advanced Projects Group for their generous support of this work. I would also like to thank Nick Chrisman for many enthusiastic conversations on this topic.

REFERENCES

- Ballard, Dana H. (1981). "Strip Trees: A Hierarchical Representation for Curves." *Comm ACM* 24, May, 310-321.
- Bentley, Jon Louis (1975). "Multidimensional Binary Search Trees Used for Associative Searching." *Comm ACM* 18, September, 509-517.
- Burton, Warren (1977). "Representation of Many-Sided Polygonal Lines for Rapid Processing." *Comm ACM* 20, March, 166-171.
- Chrisman, Nicholas R. (1988). "Spatial Indexing Schemes for GIS." Unpublished paper, Dept of Geography, Univ of Washington, October.
- Fagin, R.; Nievergelt, J.; Pippenger, N.; and Strong, R. (1979). "Extendible Hashing: A Fast Access Method for Dynamic Files." *ACM Transactions on Database Systems* 4, 3, 315-344.
- Faloutsos, C.; Sellis, T.; and Roussopoulos, N. (1987). "Analysis of Object-Oriented Spatial Access Methods." Proceedings of SIGMOD '87, 426-429.
- Finkel, R. A. and Bentley, J. L. (1974). "Quad Trees: A Data Structure for Retrieval on Composite Keys." *Acta Informatica* 4, 1-9.

- Frank, A. (1983). "Storage Methods for Space-Related Data: The Field Tree." Institut fur Geodasie und Photogrammetrie, ETH, Zurich, Nr 71.
- Freeston, Michael (1987). "The BANG File: A New Kind of Grid File." Proceedings of SIGMOD '87, 260-269.
- Fries, O.; Mehlhorn, K.; Naher, S.; and Tsakalidis, A. (1987). "A log log n Data Structure for Three-Sided Range Queries." *Inf Proc Ltr* 25, 269-273.
- Fuchs, H.; Kedem, Z.; and Naylor, B. (1980). "On Visible Surface Generation by A Priori Tree Structures." *Computer Graphics* 14, 3.
- Gunther, Oliver (1986). "The Cell Tree: An Index for Geometric Data." Electronic Research Lab, UCB/ERL M86/89. UC Berkeley, December.
- Guttman, Antonin (1984). "R-Trees: A Dynamic Index Structure for Spatial Searching." Proceedings of SIGMOD '84, 47-57.
- Herring, John R. (1987). "TIGRIS: Topologically Integrated Geographic Information System." Proceedings of Auto-Carto 8, 282-291.
- Kedem, G. (1982). "The Quad-CIF tree: A Data Structure for Hierarchical On-Line Algorithms." Proceedings, Design Automation Conference, 352-357.
- Langran, Gail and Chrisman, Nicholas (1988). "A Framework for Spatiotemporal Information." *Cartographica* 25, 3.
- Langran, Gail (1988). "Temporal GIS Design Tradeoffs" Proceedings of GIS/LIS '88 Volume 2, 890-899.
- Liou, J. H. and Yao, S. B. (1977). "Multidimensional Clustering for Database Organizations." *Information Systems* 2, 4, 187-198.
- Lee, D. T. and Wong, C. K. (1980). "Quintary Trees: A File Structure for Multidimensional Database Systems." *ACM Trans DB* 5, 3, 339-353.
- Lum, V.; Dadum, P.; et al. (1984). "Designing DBMS Support for the Temporal Dimension." Proceedings of SIGMOD '84, 115-126.
- Nievergelt, J.; Hinterberger, H.; and Sevcik, K. C. (1984). "The Grid File: An Adaptable, Symmetric Multikey File Structure." *ACM Trans DB* 9, 1.
- Noronha, V. (1988). "A Survey of Hierarchical Partitioning Methods for Vector Images." Proceedings of the International Symposium on Spatial Data Handling, Sydney.
- Orenstein, J. A. (1986). "Spatial Query Processing in an Object-Oriented Database System." Proceedings of SIGMOD '86.
- Robinson, J. T. (1981). "The K-D-B Tree: A Search Structure for Large Multidimensional Dynamic Indexes." Proceedings of SIGMOD '81.
- Rothnie, J. B. and Lozano, T. (1974). "Attribute-Based File Organization in a Paged Environment." *Comm ACM* 17, 2, 63-69.
- Roussopoulos, N. and Liefker, D. (1985). "Direct Spatial Search on Pictorial Databases Using Packed R-Trees." Proceedings of SIGMOD '85, 17-31.
- Samet, H. (1983). "The Quadtree and Related Hierarchical Data Structures." *ACM Computing Surveys* 16, 2, June.
- Tamminen, M. (1981). "The EXCEL Method for Efficient Geometric Access to Data." Acta Polytech. Scandinavia. Mathematics and Computer Science Series, 34, Helsinki.
- White, Marvin (1975). "Map Editing Using a Topological Access System." Proceedings of Auto-Carto 2, 422-429.
- White, Marvin (1987). "Digital Map Requirements of Vehicle Navigation." Proceedings of Auto-Carto 8, 552-561.

**COMPONENTS OF MODEL CURRICULA DEVELOPMENT FOR GIS
IN UNIVERSITY EDUCATION**

Timothy L. Nyerges
Assistant Professor
Department of Geography
Smith Hall DP-10
Seattle, WA 98195
U. S. A.

ABSTRACT

The tremendous growth and interest in geographic information systems (GIS) motivates a need for the development of model curricula in university education. Identifying components for model curricula development helps clarify the issues that need to be addressed. Six panelists discuss the components of model curricula for GIS in university education.

INTRODUCTION

Interest in geographic information systems (GIS) is growing exponentially as many local, state and national organizations in both government and business are investigating better ways to manage and analyze geographically oriented data with the use of computers. GIS development and use in all sectors of society motivates an examination of curricula for GIS education, especially model curricula in university education. Now more than ever, identifying components of model curricula for GIS education in universities is critical to assist in educating those individuals becoming interested in GIS, including faculty members having only limited interest in the past. The approach at the current time is on curricula components rather than a single curriculum, since GIS education exists in many different contexts. However, the goal for the future should be a curriculum from which educators could draw to develop instructional programs, and be confident that most issues in GIS education would be addressed.

This panel has been convened to discuss the components of model curricula for GIS in university education. Components involve the more conceptual issues of model curricula rather than the details of exactly what is to be done. Hopefully the latter will come at some time. As such, the focus in the panel discussion is on "education" rather than "training". Education is taken to be more fundamental and broader in scope than training which tends to focus on the use of a particular system.

All panel members have at some time written about issues involving model curricula for GIS or a closely related topic. The panelists are:

Panel Moderator:

Asst. Prof. Timothy Nyerges
Department of Geography
Smith Hall DP-10
University of Washington
Seattle, Washington 98195
(206) 543-5296

Prof. Duane Marble
Department of Geography
190 N. Oval Mall
Ohio State University
Columbus, Ohio 43210
(614) 292-2250

Assoc. Prof. James Carter
III

Computing Center
University of Tennessee
Knoxville, TN 37996
(615) 974-2418

Asst. Prof. John Morgan

Dept of Geography and
Environmental Planning
Towson State University
Towson, MD 21204
(301) 321-2973

Prof. Michael Goodchild
Department of Geography
Architect
University of California
at Santa Barbara
Santa Barbara, CA 93106
(805) 961-3663

Prof. Bernard Niemann
Dept of Landscape

University of Wisconsin -
Madison
Madison, WI 53705
(608) 263-5534

ISSUES IN MODEL CURRICULA DEVELOPMENT

Panel members have been asked to address several issues related to model curricula development. Several of these issues were identified during two recent workshops. The first was a two-day workshop held at the Ohio State University on April 30 and May 1, 1988 called "GIS in University Education" organized Duane Marble and sponsored by the IGU Committee on Geographical Data Sensing and Processing. Over eighty teaching faculty, researchers, staff, and students participated.

The second was a one and one-half day workshop organized by the author called the "Northwest International Geographical Information Systems Forum on Teaching and Research" held October 28-29, 1988 in Friday Harbor Washington. Approximately thirty faculty, staff and students from universities in British Columbia, Oregon State and Washington State as well as Michael Goodchild from the National Center for Geographic Information and Analysis attended. In some way or another all attendees at these meetings contributed to the development of this list.

In addition, several panelists provided comments and additions to the list. The following issues, and undoubtedly others, need be considered when exploring the development of model c for GIS education:

1. **Mission.** Recognition of an organization's mission with respect to teaching can be focused on: a) basic principles - service to the university, b) how to use tools in applications, and c) how to build tools. These issues need to be addressed in the context of intra-institutional departmental cooperation, inter-institutional orientation and relationship of regional cooperation with national research centers. The tasks to be addressed are: a) undergraduate education, b) graduate level education, c) extension center education, d) instructor education, and e) researcher use of GIS.

2. **Conceptual Framework.** Frameworks to help conceptualize topics and courses might be useful. Matrices might be useful to help organize discussion for missions, topics and courses. This results in three matrices, one with topics and missions as the dimensions, another with topics and courses, and another with missions and courses as the dimensions as outlined below. (Remembering that particular courses fit particular missions - more or less).

Table 1. Missions and Topics

MISSIONS (as in 1. above)
 a b c d e

TOPICS

- 1.
- 2.
- 3.
- 4.
- 5.
- etc.

Table 2. Courses and Topics

COURSES
 first year second year
 1 2 3 4 5 6

TOPICS

- 1.
- 2.
- 3.
- 4.
- 5.
- etc.

Entries in Tables 1 and 2 would represent a certain depth of presentation and expected outcome in terms of understanding a topic. The levels can be: 1) exposure to topic 2) understanding of principles behind topic 3) use of tools 4) able to build own tools. An approach like this has been taken in (Nyerges and Chrisman 1989) to develop an integrated instructional program in computer-assisted cartography and GIS.

The entries in Tables 1 and 2 can be used to generate a summary of courses such as depicted in Table 3.

Table 3. Courses and Missions

		COURSES					
		first year			second year		
MISSIONS		1	2	3	4	5	6

- a.
- b.
- c.
- d.
- e.

3. **Prerequisites.** Identification of the prerequisite courses for a GIS program can be performed only after the topics in issue 2 above have been documented. Courses and topics in mathematics, computer science, mapping sciences with perhaps others must be addressed.

4. **Course Integration.** What is the appropriate mix of integrating mapping sciences with GIS, especially cartography and remote sensing?

5. **Tutorials.** Reduction of the amount of startup time for students to learn a concept is important. Having the appropriate tutorial environment can be very important. What is the appropriate length of time with regard to:

- a) user interface learning
- b) database development

6. **Software/hardware Tools.** A need for pedagogic tools is evident. Tools are needed for a) demonstration of principles, b) tools for use in GIS project development, and c) tool building. Different software and hardware might be required to suit the general needs of different program orientations. A list of the functionality of such tools that are available would be necessary to satisfy the needs of these orientations. A list of could be useful, but more than one tool might be required to satisfy the orientations.

7. **Regional Cooperation.** The basis for developing cooperation within and among institutions for offering courses needs to be explored. Is it useful to develop regional forums for discussion of GIS teaching and research issues? Local funding for training and research could enhance programs.

8. **Laboratory Funding.** A need exists to develop a collective statement about problems with the funding of laboratory space. Perhaps several case studies can be developed that describe laboratory maintenance. Staffing, software, hardware, and data maintenance should be included.

9. **Balance of Theory and Application.** The best way to deliver a theoretical message in an application setting is in need of exploration. Identify the appropriate mix of theory and application. This depends upon the orientation and level of the course. Perhaps a balance as suggested in Tables 4 and 5 might be appropriate.

Table 4. Topic Balance for Concept Presentation and Tool Use Instruction

Tool Use <u>Level</u>	<u>Theory</u>	<u>Application</u>
Beginning	25%	75%
Intermediate	25%	75%
Advanced	25%	75%

Table 5. Topic Balance for Tool Building Instruction

Tool Building <u>Level</u>	<u>Theory</u>	<u>Application</u>
Beginning	25%	75%
Intermediate	50%	50%
Advanced	75%	25%

10. **Standard Data Sets.** Standard data sets would help with the delivery of fundamental issues in a tool use environment and testing of software/hardware in a tool evaluation environment.

11. **Course Linkages.** Identification of the linkages with topics in social science, environmental science, physical science, mapping sciences, etc. is needed to broaden the perspectives of students. Integrating GIS with these topics can prove to be demanding, but necessary, to provide students with a framework that goes beyond the tools. This might be a difficult task because it is often idiosyncratic to any given instructional program.

12. **Learning Environments.** Determining the importance of a collegial community in the learning process of GIS, e.g. group work sessions and discussion sessions can be important in the delivery of instruction.

13. **Instructional Evaluation.** Identification and preparation of evaluation criteria for GIS instruction can be useful as instructional, performance indicators.

SUMMARY and CONCLUSION

Several model curricula development issues must be considered for GIS in university education. Several of these issues have been presented in the panel session and others will surface as discussion continues on this topic. Perhaps the most effective way to proceed is to identify instructional missions, set goals and to then identify

topics to be included in the instructional program. These topics could be developed using the mission by topic and course by topic frameworks presented in Tables 2 and 3, respectively. Filling in the matrices involves identifying an appropriate level of exposure to a topic for introductory, intermediate and advanced courses.

The National Center for Geographic Information and Analysis has been preparing a three course sequence which perhaps can be used as the initial ground work for a model curricula development. Prerequisites, laboratory environments and cognate courses need more directed discussions than can be accomplished in forums such as a conference panel session. Discussions on these topics can only be effective through broad-based participatory effort on the part of the professional societies involved. Hopefully the issues discussed by this panel can continue in the future through more directed efforts.

REFERENCES

Nyerges, T. L. and N. R. Chrisman 1989. A Framework for Model Curricula Development in Cartography and Geographic Information Systems, *Professional Geographer*, in press.

AUTOMATED NAMES PLACEMENT IN A NON-INTERACTIVE ENVIRONMENT

Lee R. Ebinger and Ann M. Goulette
Geography Division
U.S. Bureau of the Census
Washington, DC 20233

ABSTRACT

Accurate and aesthetic placement of text is an important component of a well-designed map. Census Bureau cartographers and computer programmers have worked together to incorporate cartographic names placement conventions into mapping software. The result is an effective names placement system that is capable of positioning text and resolving text placement conflict without the need for human intervention. The names placement software is part of the automated mapping system developed to provide paper maps for numerous 1990 Decennial Census activities. This paper examines the current automated names placement system and describes algorithms for labeling point, line and area features. Several approaches for handling text conflict and text storage are also discussed.

INTRODUCTION

Poorly placed names are often the identifying factor of a computer-generated map. Even with sophisticated algorithms, computers have yet to produce maps with text placement that matches manual placement by trained cartographers. Computer-generated maps have been noted for overlapping text, upside-down text and text placed at awkward angles -- all properties that cartographers strive to avoid. As a result, some computer-generated maps have used predetermined coordinates, interactive techniques or a manually produced type stickup as an overlay to the computer-produced base map for placement of labels. These solutions lead to maps that are more accurately called "computer-assisted" rather than "computer-generated." Despite great improvements in other areas of computer cartography, automated names placement remains a problem.

The rules for manual label placement can be found in any introductory cartography textbook (e. g., Robinson, et al., 1978) and have been documented by Imhof (1975). Imhof's fundamental rules are that names: 1) should be legible; 2) should be easily associated with the features they describe; 3) should not overlap other map contents; 4) should be placed so as to show the extent of the object; 5) should reflect the hierarchy of objects by the use of different font styles; and 6) should not be densely clustered nor evenly dispersed. Imhof asserts that these rules are often violated because satisfying one rule may break another.

Research in automated names placement has acknowledged text placement conventions and the difficulty involved in fulfilling the established rules. Monmonier (1982) states that the goal of automated names placement is to optimize the number of placements that follow

cartographic conventions and guidelines. Zoraster (1986) allows for deletion of labels that would overlap if placed. Other researchers (Ahn & Freeman, 1983; Cromley, 1983; and Pfefferkorn, et al., 1985) attempt to place names that have the smallest degree of freedom first and those that are less constrained in position afterward. All the aforementioned research use a recursive process that repositions as many labels as necessary to find appropriate locations for all text and encourages human intervention as the final step to improve label positions.

THE PROBLEM

Recursive computer processing and human intervention are not feasible solutions for the automated names placement problem at the U.S. Bureau of the Census due to the large volume of maps produced. For the collection of data for the 1990 Decennial Census, the Census Bureau will produce through automated means over ten different map types with nationwide coverage. This effort will produce an estimated one million different map sheets on monochromatic electrostatic plotters. All of the maps are highly dependent on accurate names placement and use guidelines compiled from accepted cartographic conventions and census traditions for the placement of labels. All maps must be plotted under rigid deadlines using limited computer and staff resources. Given these constraints, inefficient use of computing time and dependence on human map inspection for improved names placement are not possible.

There are many effective ways to improve label positioning without resorting to manual intervention or recursive processing. For instance, the map scale can be increased or larger scale insets can be made in areas of feature density. Text size can be decreased. Text may be repositioned within or along the feature or object. A text placement priority by feature type may be established to minimize names conflict. Text may be angled, hyphenated, stacked or interletter spaced. Arrows may be used in difficult cases to associate text with its object in a congested area. Fishhook symbols (see Block 124 in Figure 1) may eliminate the placement of duplicate names of adjacent areas with the same identifier. Text may be replaced by key numbers and a key listing. As a last resort, text may be suppressed altogether.

The Census Bureau employs some of these techniques in its completely automated names placement algorithms. Maps are preprocessed to determine adequate scale and to identify rectangular windows for insets of larger scale. However, most map types have practical constraints on the maximum number of sheets and sheet sizes. On some map types, text size may be decreased as long as legibility is preserved. All maps utilize a predetermined order to place feature names in terms of their importance to the map. The software provides for multiple alternative placement positions, as well as for stacking text, and in some cases, angling text to improve placement. The use of arrows and fishhooks, elements of Census Bureau mapping convention, has been incorporated into the names placement routines. Because naming features is crucial to census maps, suppression of labels is used only as a final alternative. Interletter spacing, hyphenating words and key numbering are currently not a part of the software.

The automated names placement algorithms used to produce the maps required for 1990 Decennial Census data collection represent a real-world application of a non-interactive, non-recursive system. The algorithms place labels for points, lines and areas. Overlapping text is avoided, although some overlap is allowed through the use of "see-through" screened fonts. Alternative placements are attempted before more radical procedures (arrows, fishhooks and suppression) are implemented. Text is not always placed as a manual cartographer would position it; however, the algorithms have resulted in readable, effective maps for 1990 Decennial Census operations.

THE ALGORITHMS

An integrated cartographic text placement system must incorporate algorithms for positioning names of point, linear and areal features. The problem of names placement differs significantly for the three types of data. The Census Bureau approach to point names placement is rather simple; the approach to linear and areal names placement is more sophisticated. In addition to these algorithms, the system must provide a method for the detection of text overlap. All of the names placement algorithms use an overlap detection routine for determining the final position of labels. Figure 1 shows examples of many features of the names placement algorithms.

Point Names Placement

The point names placement algorithm is used to identify point features depicted by pictorial symbols. The point symbol is centered on a single coordinate and the name is offset from the symbol. The algorithm begins by testing the position of the symbol for overlap with other text and symbols. If conflict is detected, neither the symbol nor the label is plotted. Otherwise, one of four ranked positions for the label is tested for overlap until a non-overlapping location is found. These positions are: 1) above and to the right of the symbol; 2) below and to the right; 3) above and to the left; and 4) below and to the left. If the name cannot be placed at any of the four positions, only the point symbol is placed. The name is always placed parallel to the horizontal axis.

Linear Names Placement

The linear names placement algorithm is used to label linear features such as roads or streams. Most often, linear features are labeled with a name or code, but on some map types, street address ranges are plotted also. Address ranges present a more difficult problem: they must be placed on the correct side of the street and reflect the relative direction in which the range increases along the street segment. In implementing the algorithm, the following guidelines are used: 1) text is placed right-reading and follows the curvature of the feature; 2) text is not allowed to overprint the linear feature; 3) text is not permitted to overlap previously placed text and, in some instances, point symbols; and 4) text placement maintains a true ground-to-map positional and directional orientation for address ranges. As in the point-oriented algorithm, when text conflict is detected, alternative positions are examined. If no suitable placement is found, the algorithm provides a user-defined option to force the text to plot despite overlap or to suppress the label.

The first procedure of the algorithm checks whether the feature consists of one or more segments. Different methods are used for single and multiple segment features. For single segment features, the segment length and text length are compared. If the segment length is not sufficient, multiple word names may be stacked and placed above and below the segment. Alternatively, the name may be placed on one line above the segment. Address ranges may be stacked or centered along the segment. As a last resort to placement, text is allowed to extend beyond the end of the line segment.

If the feature consists of more than one segment, the algorithm first attempts to place the text on the longest segment, provided the segment length exceeds the text string length. If the length constraint is met but overlap with other text is encountered, the name is moved incrementally along the segment until a non-overlapping position that meets the length requirement is found. If moving the text is unsuccessful, the procedure is repeated for the next longest segment. When the procedure has exhausted all possible single segment placements, multiple segments are tried. In these cases, the algorithm selects the two longest consecutive segments and

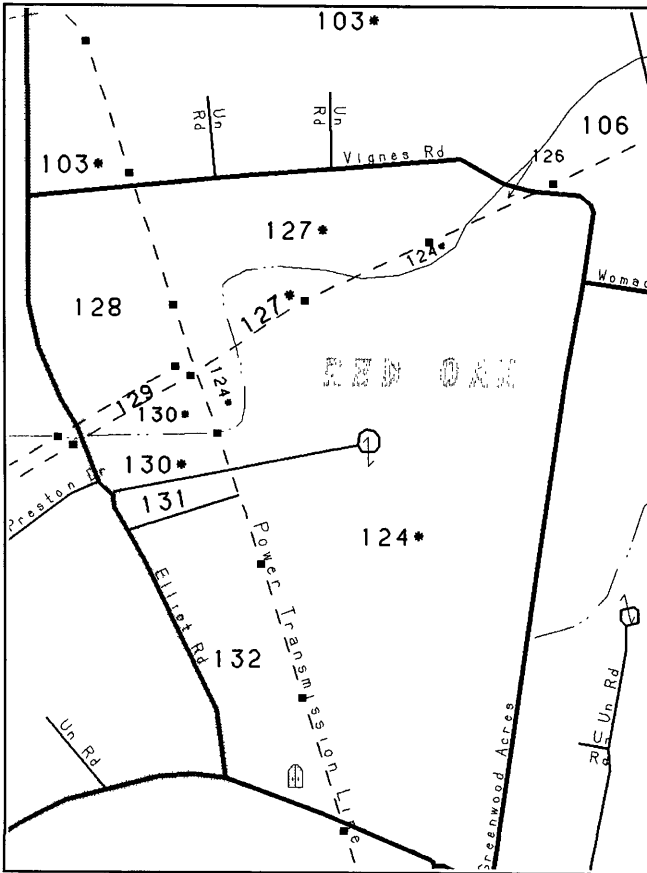


Figure 1. Portion of a data collection map with examples of linear and areal names placement using the non-interactive placement algorithms.

tries to place the name if the length of the combined segments exceeds the text length. Text is broken into substrings for each segment. The substrings are placed at angles corresponding to the angles of the segments. If the name cannot be placed, the process is repeated using the three longest consecutive segments and so on. If all segments have failed the procedure, the name is placed along and beyond the feature using all segments of the feature. Address ranges are restricted to a single segment of the multiple segment feature. The address range may be stacked or centered along the line segment.

After the number of segments is determined, a position for the text is calculated and checked for overlap each time text placement is attempted. If the position is unsuitable, another position is selected and tested. The placement, orientation, and test for suitability of placement are performed in three steps: finding the starting node of the selected line segment(s), calculating the lower-left corner coordinate of the text rectangle and detecting text overlap.

First, it is necessary to locate the starting node of the portion of the feature where the name is to be placed, given the constraint that text must be right-reading. The segment(s) used for placement has both a starting and ending node. The leftmost of these nodes is selected as the starting point for text placement. For vertical lines, the topmost node is the starting point. Although successful for most cases, this method may occasionally fail for sinuous line segment chains.

The lower-left corner of the label is determined next. The text is always offset diagonally from the starting point (above or below the line segment and toward the end of the segment) to avoid obscuring information. For vertical lines, the offset is always positive.

Finally, the coordinates of the text rectangle are calculated and the rectangle is checked for overlap with previously placed text and point symbols. If the text is a substring, it is also checked against other members of the text string. If no text overlap is detected, the text rectangle position is stored and the text string is plotted.

The aesthetic placement and legibility of linear feature names are affected by the map scale. On small-scale maps, the curvature of linear features is more extreme than on large-scale maps. As a result, text placement on large-scale maps is more cohesive. Although aesthetic placement could be improved by smoothing the coordinates of the underlying feature, this is not implemented in the current algorithm. Small-scale maps also have higher feature density. This causes small-scale maps to have a higher ratio of unlabeled-to-labeled features because of increased text conflict. In the current algorithm, conflict between text and most features is not detected. Conflict detection is performed only against previously placed text and point symbols. Therefore, text is easily obscured by a dense feature network.

Areal Names Placement

The areal names placement algorithm is used to label polygons such as water bodies and political or statistical areas. The algorithm finds a location within a polygon where a name can be placed parallel to the horizontal axis. Should text conflict occur, the algorithm provides several options for manipulating the text so that other alternatives are available.

The areal names placement algorithm performs two major functions. The first function heuristically selects points within a polygon to be considered as a center for the text. The second function determines the suitability of the point locations and provides placement options should the original point location be unsatisfactory for text placement.

Providing multiple potential coordinates for text placement within a polygon allows greater flexibility for the final placement of text. The optimal position is one where the text fits entirely within the polygon, is free of text conflict, and is near the center of the polygon. Using a scan-line technique, potential text locations are selected using two criteria: 1) the centrality of the coordinate within the polygon, and 2) the length of a horizontal scan-line segment that intersects the edges of the polygon. Centrality describes the proximity of a point to the center y-coordinate of the polygon. The length of the scan-line segment refers to the space available to place a name horizontally within the polygon. The length is calculated as the difference between the minimum and maximum x-coordinates of the line segment. A single scan-line can yield multiple line segments if the subject polygon has internal polygons (islands) which break the scan-line into smaller, discontinuous segments. The number of scan-line segments calculated for each polygon is specified as a parameter in the algorithm.

The areal names placement algorithm orders scan-line segments based upon a criterion of centrality, segment length, or a combination of segment length and centrality, as specified for the algorithm. First, the subject polygon is dissected by the specified number of scan-lines at equal intervals to obtain the horizontal line segments. The line segments are then sorted by the chosen criterion. By selecting centrality as the criterion, the longest horizontal line segments with a unique y-value are collected and reordered by their proximity to the center y-coordinate of the polygon. The line segment ordering is from middle to top and bottom using alternating scan-line positions. The sort by length option orders the line segments from the longest to the shortest segments. The combined length-centrality sort first performs a length sort on the line segments, then the longest line segments are ordered as to their centrality. Figure 2 illustrates the result of the scan-line method.

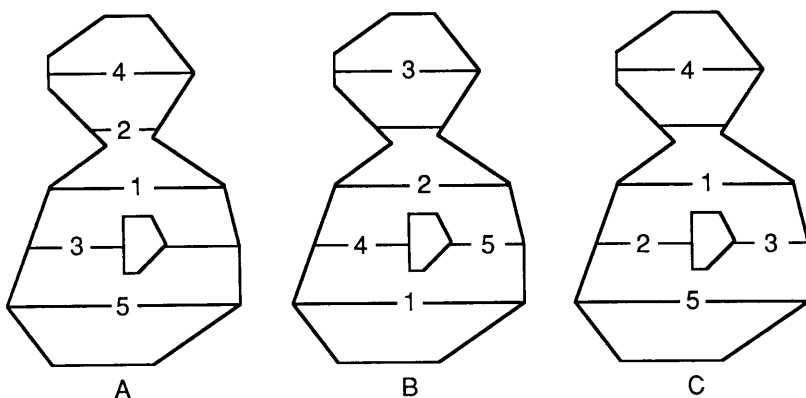


Figure 2. Different results of the scan-line text placement algorithm based on sorts by (a) centrality, (b) segment length, and (c) a combination of segment length and centrality. The numbers represent the priority of the scan-lines for text placement.

The areal names placement algorithm centers the text along a scan-line segment and tests for text conflict. An optional test is performed if the name must fit entirely within the polygon. When either test detects a problem with the text placement, the algorithm uses other options to manipulate the text.

Text for areal features may be plotted on one, two or three lines. Words on multiple lines are arranged to minimize the difference in the number of characters per line. All lines of text are centered around the midpoint of the scan-line segment. In addition, each line of text is checked for potential text conflict. If no conflict-free position is found, the algorithm provides the option to force the name to plot or to omit the name.

For some types of areal features, such as census blocks, the identifier must be located completely within the polygon. Two checks are performed upon the text position: a point-in-polygon check and a line-intersection check. Prior to the checks, a rectangle is constructed around the text to be used to approximate the text position. In the point-in-polygon test, the corner coordinates of the text rectangle are tested to determine whether they lie within the interior of the polygon. In the line-intersection test, the four line segments forming the text rectangle are checked to ensure that they do not intersect with the edges of the polygon. If either of the checks detects an overlap condition, an alternative to this text position is found.

Given that polygons appear in a variety of shapes, placement of text at a specific location may not be acceptable. The areal names placement algorithm includes several options for resolving the problem of text placement. For example, the text position may be rotated so that it is parallel to the longest line segment of the perimeter of the polygon. The text position may be offset from the original coordinate. The size of the text may be reduced. A different scan-line segment midpoint may be used. An arrow may be used to associate text placed outside the polygon with the center of the polygon. Unlabeled polygons may be fishhooked to adjacent polygons of the same name.

Depending on the map type, these options have different priorities. Generally, it is sufficient to select a different scan-line segment midpoint or reduce the text size in order to find an acceptable position for the text. The other options listed are applied if a more rigorous search for a solution to the text placement problem is required and if the options are relevant to the map type.

The areal names placement algorithm performs well if the shape of the polygon is not too irregular and the text rectangle is small compared to the area of the polygon. Text is placed on a straight line; the algorithm does not attempt to curve the name to conform to the shape of the polygon. For some areal features, such as small lakes or towns, the text size frequently exceeds the size of the polygon. These names are not arrowed to the polygon, since offsetting the name outside the polygon consumes space elsewhere on the map. In this case, the name is simply centered in the polygon and allowed to extend beyond its edges.

The use of screened fonts has proved advantageous for identifying some types of areal features. These fonts allow text to overprint other features and labels without obscuring information.

Detection of Text Overlap and Text Storage

The ability to detect text conflict and store text positions is essential for producing a legible map. Two methods have been developed that use similar procedures for detecting text overlap, but differ in the way text positions are stored. One method, a vector method, stores positional information in the form of coordinate values at map scale. The other method, a grid-based procedure, converts the positional data into row and column values. Both methods produce similar results in terms of the number of text conflicts detected and the amount of processing time required.

In order to efficiently perform text conflict detection and text storage, a text rectangle is used to approximate the position and extent of the text string. In addition, a second rectangle is constructed from the maximum and minimum x-y coordinates of the text rectangle. This second rectangle, the text envelope, is used to streamline text conflict detection.

The vector method calculates coordinate values at map scale for the text rectangle and envelope. These coordinate values and the angle of the text rectangle are stored in the memory of the computer as real numbers.

The grid method superimposes a grid over the map. The location of a grid cell is defined by row and column positions. This method stores the positional information of the envelope surrounding the text rectangle by determining two parameters: the column and row numbers of the grid cell that correspond to the upper-left corner of the text envelope and the number of rows and columns of the grid cells occupied by the text envelope. Grid cell locations of the text rectangle are determined and stored also. If the text rectangle coincides with the centroid of a grid cell, that grid cell becomes off-limits to subsequent text placement. Row and column information is packed into bit positions and stored in memory.

In both methods, the detection of text conflict consists of two checks. The first check tests whether the current text envelope overlaps any previously stored text envelopes. When text envelopes overlap, a more refined check is performed using the data stored for the text rectangles. If no overlap of text rectangles is detected, the positional data for the current text envelope and rectangle is stored.

The methods differ, however, in implementation of the checks. The vector method compares the maximum and minimum text envelope values. If the current envelope overlaps a stored envelope, a line intersection test determines whether the text rectangles overlap. The grid method tests the text envelopes by comparing the beginning and ending grid cells of the current label to the stored text envelope grid values. If a more rigorous check is required, the grid cells occupied by the text rectangles are tested for matching row and column values.

For linear names placement, an additional check for text conflict is performed. If a text string is broken into substrings in order to follow the curvature of a multiple-segment line, each substring must be checked for text overlap against other members of the text string. If any substring causes overlap, the entire string must be repositioned. The vector method processes substrings consecutively and stores the positional information before proceeding to the next substring. The linear names placement

algorithm keeps track of the number of text substrings stored. If text overlap occurs while processing a text substring, the linear names placement algorithm deletes the information for each substring from the storage array and begins again. The grid method uses a different technique. When consecutive characters are closely spaced along a curved line, the proximity and angularity of the text rectangles cause frequent overlap of the same grid cells. Instead of using grid cells to represent the position of the text string, the characters of the text string are inscribed within circles. The distances between circle centroids are measured and compared to a specified distance that allows a minor degree of overlap. This internal test on the text string is performed after testing the substring against stored text using the grid cell method.

The application of either storage method has advantages and disadvantages. Factors involved in evaluating the relative performance of the two methods include the amount of memory necessary for the application and the effectiveness of the names placement.

Restricting the size of arrays is important for keeping the overall size of the mapping programs below the maximum limit permitted by Census Bureau hardware. The vector method uses twenty-six 36-bit words of memory to store the positional information for a text rectangle. The raster method generally consumes less space by packing integer values into bit positions. The number of words stored depends upon the angle, length and height of the text rectangle. Because the vector method uses more space, it can store fewer text positions. Therefore, large-scale maps containing less text can use the vector storage method; small-scale maps with greater text placement requirements must rely on the grid method for storing text positions.

The grid method has two major drawbacks. First, text overlap cannot be detected where portions of the text rectangle fall within a grid cell but do not fall on the centroid. For these cases, the line intersection test of the vector method is superior for detecting text overlap. Second, there is a limit imposed upon the grid dimensions and size of the text rectangle by the number of bits in the word used to store grid cell values. These limits restrict the map image area size, text length and text height. The current image size limitation is approximately 41.0" in both dimensions; either text length or height is limited to 10.2". The vector storage method sets no practical limits upon map size and text dimensions.

Currently, the grid method has an important advantage over the vector method. Normally, text is not permitted to extend beyond the map borders or appear within inset areas. The grid method allows the map image area dimensions and rectangular inset windows to be defined precisely. When text is checked for overlap, the text position can be compared against these limits, confining text placement to the map image area. This check is not implemented in the vector procedure, but it may be included in future software.

The most effective overlap detection algorithm would assimilate the best aspects of both methods. It would be capable of accurate text conflict detection and efficient text storage. In addition, it would use a minimum amount of computer processing time and allow as much text placement as possible in the most acceptable positions. Revisions to the software will attempt to optimize these features.

SUMMARY

The automated names placement algorithms described here represent the efforts of Census Bureau cartographers and computer programmers over several years of names placement guidelines generation, software development and refinement, and map production. As much as possible, these efforts have attempted to replicate, given hardware and time constraints, conventional placement of names on maps. Through an on-going critical review process, the software has evolved into its present state. The result has been the effective placement of names for point, linear and areal data without resort to interactive repositioning or recursive processing. Although some label placements are awkward, the majority of text is positioned using established cartographic rules.

The Census Bureau intends to further refine its automated names placement algorithms to improve the appearance of text and to extend the algorithms to maps produced for publication purposes. For instance, improvements to the names placement algorithm may include the ability to erase and reposition text, repeat the label along the feature, break the text string by syllables, and smooth the underlying feature. The capability to detect text overlap against all linear and areal features may be added. In addition, the ability to confine text to the image area limits may be added to the vector method for text overlap detection. For publication maps distributed to the public, more aesthetic enhancements, such as key numbers, may be included. Automated names placement on these maps may be aided by interactive editing.

REFERENCES

- Ahn, J. & Freeman, H. 1983. A program for automated name placement. Proceedings, Auto-Carto VI, pp. 444-453.
- Cromley, R.G. 1983. An LP relaxation procedure for annotating point features using interactive graphics. Proceedings, Auto-Carto VI, pp. 127-132.
- Imhof, E. 1975. Positioning names on maps. The American Cartographer, Vol 2, No. 2, pp. 128-144.
- Monmonier, M.S. 1982. Computer-Assisted Cartography Principles and Prospects. Prentice-Hall, Inc., Englewood Cliffs, NJ.
- Pfefferkorn, C.; Burr, D.; Harrison, D.; Heckman, B.; Oresky, C.; & Rothermel, J. 1985. ACES: A cartographic expert system. Proceedings, Auto-Carto VII, pp. 399-407.
- Robinson, A.; Sale, R.; & Morrison, J. 1978. Elements of Cartography. John Wiley & Sons, New York.
- Zoraster, S. 1986. Integer programming applied to the map label placement problem. Cartographica, Vol. 23, No. 3, pp. 16-27.

AN EXPERT SYSTEM FOR DENSE-MAP NAME PLACEMENT

Jeffrey S. Doerschler
Hamilton Standard
Division of United Technologies Corp.
Windsor Locks, CT 06096

Herbert Freeman
CAIP Center
Rutgers University
Piscataway, NJ 08855-1390

ABSTRACT

A dense map consists of closely-spaced topographical and political features labeled by aesthetically placed names. Most existing systems designed to place names automatically are limited because they label lower-level points and lines, not higher-level features. To overcome these limitations, a fully automated rule-based cartography system was developed to label higher-level features on dense maps. The feature-based approach allows the cartographer to create independent spatial data that may partially define several features; to create different thematic maps from a single database; to define relationships among classes of features; and to determine how features will be represented and labeled.

INTRODUCTION

A map is a collection of topographical features within a geographical region. These features may be represented on the map by symbols, lines or boundaries. Names on the map identify and classify features and relate them to their physical counterparts.

An automatic map-production system must be able to access a digital map database, select the features that are to appear to be on a map, annotate features and produce maps in a human-readable form. Placement of names on these maps is one of the more complex and difficult processes to automate. The difficulty depends to a large extent on the density of the map. On sparse maps, there are many empty spaces where names may be placed. On dense maps features are close together, leaving little room for names.

Most previous attempts at automated name-placement have been limited to labeling point features. Yoeli (Yoeli 1972) accomplished this by dividing a map into a dense grid with each character being placed into a unique cell. Others (Kelly 1980, Hirsh 1982, Langran 1986) used different techniques to improve point feature name placement.

Name placement is much more complex when point, line, and area features need to be labeled. Several papers (Basoglu 1982, Ahn 1983, Freeman 1984) describe the development and operation of a program to label maps representing these three types of features.

Most existing name-placement systems represent and label points, lines, and areas differently; however, the complexity and variety of maps that each can represent and label is limited. Some of these systems are limited by the types of features that can be represented, by the number of features that can be represented by a single point or line, by the number of lines that can intersect, or by the ability to label features represented by different symbols or line types.

A feature-based system has been developed to demonstrate one approach to automated dense map name-placement. This system uses rules based upon those enumerated by Imhof (Imhof 1975) to determine the location, orientation, font, size, and slant of each character in feature names. After each rule is processed, the name-placement quality is measured to determine whether additional rules must be processed. If the quality of a name placement deteriorates after other names are placed, backtracking occurs and the name is repositioned.

THE DATA STRUCTURES

The major data structures used by the name-placement system include tightly-closed boundaries, k-d trees and map databases which contain topographical features. These databases are used during all phases of map production. The data comprising the features consist of spatial data, classifications, names and abbreviations, miscellaneous data, and character placement information as shown in Fig. 1.

The spatial data are points, lines, and boundaries which may describe the location, size, shape and extent of features. Many features share common points or lines. For example, a line may represent both a river and a political boundary. Spatial data is independent of feature data. It may, however, be used as part of the definition of the feature. All spatial data is stored in units of longitude and latitude so that the database will be independent of any map projection.

Each feature may have one name and an unlimited number of abbreviations, one of which will be placed on the map. Only non-standard abbreviations need to be entered, since the labeling algorithms can automatically abbreviate common words such as "N." for "North".

The classification of a feature (e.g. river, city or interstate highway) determines if the feature will be placed on a map, how it will be represented, and how it will be labeled. Other classification-dependent information may also be required to properly place names. For example, city populations may be needed to determine proper character sizes.

The last feature attribute in the map database is a list of character placements consisting of the characters in the feature name, the longitudes and latitudes of their locations, and the orientations, font, height, width, and slant of the characters. Character placements can only be determined after the map projection, style and size have been determined.

When a map is plotted, area features are often represented by solid regions drawn as a series of horizontal lines. To generate these lines, the boundary of the area is first converted to a tightly-closed boundary (Merrill 1973). Pairs of points on the boundary are grouped to form endpoints of horizontal lines. These horizontal lines or y-partitions may then be plotted or used by the name-placement algorithms.

K-d trees are used by the name-placement system to detect overlapping names and symbols. A k-d tree is a structure designed for efficient multi-key searches, including exact match, partial match, and region queries (Bentley 1975). K-d trees are formed for character and point locations. Longitudes and latitudes are used alternately on each level of the tree as keys. The leaves of the trees point to corresponding entries in the map database.

NAME-PLACEMENT SYSTEM

To create a map, a cartographer first selects the types of features to be drawn on the map. A set of rules is used to determine how each feature is to be labeled. After a name is placed, the aesthetic quality of the placement is measured, for example, by checking whether the name overlaps any existing feature or name. If the first position is unacceptable, another position is tried.

Data Files

The name-placement program uses a set of rules to determine the location of each character to be placed on a map. These rules are defined in data files to allow the cartographer both to specify the name-placement styles for each feature classification and to show relationships among these classifications. Other data files specify the order in which the names will be placed on the map, the sequences of rules to be used for each feature classification, how names are to be placed, and how their quality is to be evaluated. A block diagram of the name-placement data files is shown in Fig. 2.

The first six data files assign names to the placement algorithms. Each file contains lists of algorithm names and defines parameters required by the algorithms. Each algorithm

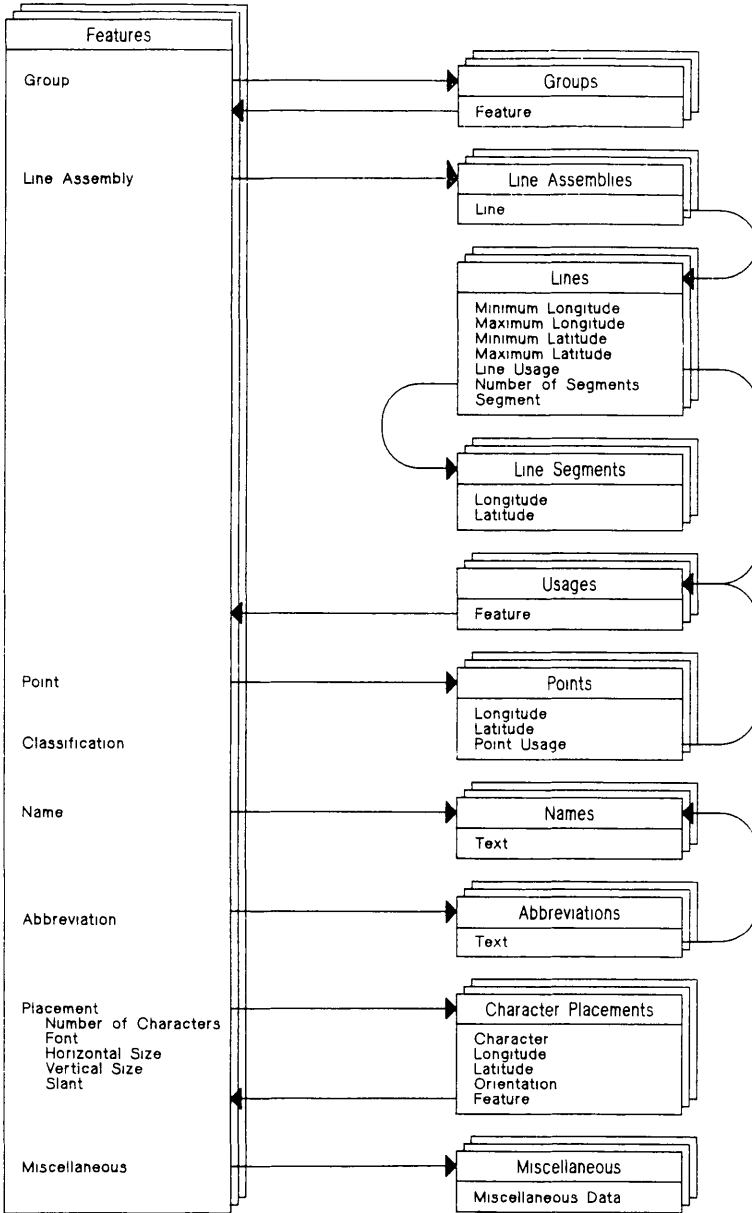


Fig. 1: Map Data Structures

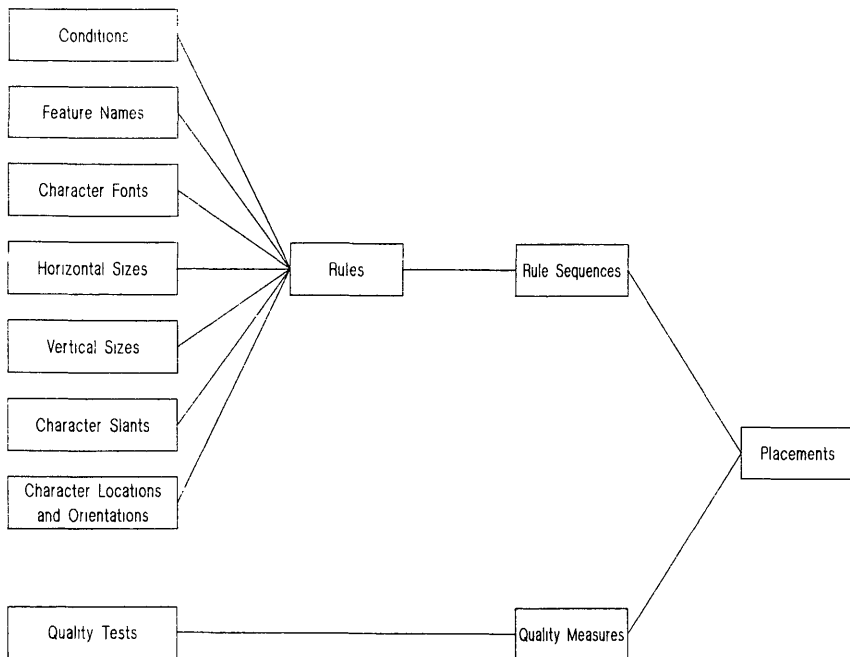


Fig. 2: Name-Placement Data Files

determines the value of one or two components of a name placement. A typical rule which uses these algorithms is the following:

Rule 101

```

Letters
  Name_with_Capital_Letters_based_on_Population 25000
Font
  Uniform_based_on_Population 1100 [Sans.1] 1101 [Sans.2] 8000
Horizontal_Scale
  Population 8 1000 10 5000 12 8000 13 25000 15 100000 18
Vertical_Scale
  Population 9 1000 10 5000 13 8000 14 25000 16 100000 18
Slant
  Uniform 0
Location
  Point_Upper_Right_Horizontal 5
  
```

In this example, rule 101 contains a list of name-placement components followed by the algorithm and its parameters which will be used to define the component. For example, the horizontal size of the characters will be defined by the algorithm "Population". The parameters to this algorithm are population intervals and character sizes in 0.1 mm units.

After a name is placed, one or more of the quality measurement algorithms are used to determine if the placement is acceptable. The names of these algorithms are defined in one of the data files whereas the set of quality measurement algorithms to be used for each feature classification and the importance of each algorithm are defined in the quality measurement data file.

If the quality of a name is unacceptable, the next rule must be tried. The rule sequence data file contains a list specifying the order in which the rules are to be tried. The last data file indicates the set of rule sequences and quality measures to be used for each feature classification.

Rule Processor

After the data files have been read, the name-placement rule processor determines the features to be labeled and the rules to be evaluated. The rule processor evaluates the first rule for each feature, possibly placing names on the map for every feature. The name placement quality will then be incrementally improved as additional rules are evaluated.

When names are placed on a map, their quality is evaluated to provide feedback to the name-placement rule processor. If a placement is unacceptable, additional positions are tried. To determine if a name placement is acceptable, its quality is evaluated using the rules specified in the quality measurement data file.

Each feature classification has a quality threshold between 0.0 and 1.0. If the quality is greater than or equal to this threshold, the name placement is acceptable. If it is less than the threshold, the placement is unacceptable. The threshold is lowered each time a rule is tried so that a placement which is initially unacceptable may become acceptable if a better placement cannot be found easily. The amount by which the threshold is lowered is specified in the rule sequence data file.

An existing name placement may, at times, have to be moved to make room for another name. This condition, known as backtracking, is required whenever a new name placement causes the quality of an existing placement to become unacceptable. After a name is placed, the quality of nearby names which may have been affected is reevaluated. If a placement is unacceptable, existing names may be repositioned.

Placement

The components of a name placement include the characters to appear on a map and their locations, orientations, vertical sizes, horizontal sizes, slants, and fonts. When a name-placement rule is evaluated, several algorithms are executed, each defining one or more of these components.

The first name-placement component that should be defined is the name or abbreviation to be placed on the map. Next, several algorithms can be used to define the character font, horizontal size, vertical size, and slant. The algorithms may determine the value of these parameters based upon the feature classification, importance of the feature, amount of room on the map or by some other characteristic such as a population.

Locations and orientations are the most difficult name-placement components to define because of the large number of possible positions, restrictions, and interactions. These two components are defined by the same algorithms because they are interdependent. The location of a character is specified by its longitude and latitude; its orientation is specified by the number of degrees it has been rotated.

Algorithms which label point features may place names next to the feature in one of eight directions or shift the name horizontally or vertically until an acceptable position is found. Long names may be split into multiple lines before they are placed on the map. Depending upon its location, a multi-line name will be left-justified, right-justified, or centered. The multi-line algorithms are illustrated in Fig. 3.

Route numbers are names which are centered horizontally on a highway. These names may also be shifted to avoid overlapping other characters, symbols, or intersections. Route numbers should normally be placed in a county, state, federal, or interstate highway symbol. Since the highway symbol and route number need to be at the same location, both are positioned by the name-placement program. A symbol is placed into the map database as if it were a character.

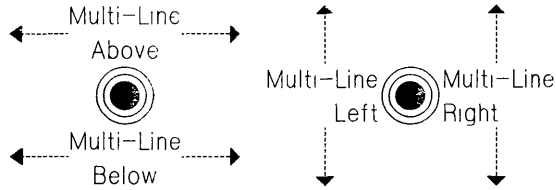


Fig. 3: Point Feature Multi-Line Name Placement

Other line features may be labeled by parallel names centered on or offset from the line. Once the name has been placed, it may be shifted to avoid overlapping existing names, existing features, map borders or sharp bends in the line. These names may also be shifted beyond the end of the line or repeated at predetermined intervals.

Many algorithms have been developed to place names within area features, along area boundaries, or adjacent to areas. A simple algorithm places a horizontal name in the middle of a feature. More complex ones will shift the name horizontally and vertically until an acceptable position is found. If the area feature is too small, the name can be placed next to the feature. Some of the same techniques used to place names next to symbols are also used to place names next to area features.

Other algorithms place names along the major skeleton of area features. A skeleton is the locus of points inside a region which is equidistant from the two or more closest distinct points on the boundary. The major skeleton is the longest continuous line in the skeleton as shown in Fig. 4. These algorithms may place a name in the center of the skeleton or shift the name along the skeleton until an acceptable position is found.

Finally, names of some area features, such as political subdivisions, should be placed along boundaries. These names will be repeated wherever a neighboring feature shares the boundary. Area feature boundary algorithms are illustrated in Fig. 5.

If none of the preferred name-placement algorithms produce an acceptable placement, additional steps can be taken. The size of the name could be reduced or the name could be eliminated from the map entirely.

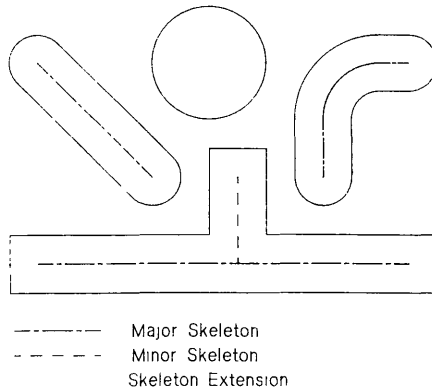


Fig. 4: Skeletons

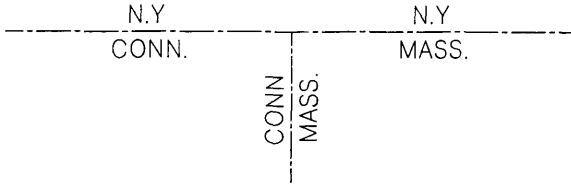


Fig. 5: Area Feature Boundary Name Placement

Quality

Once names have been placed on a map, their aesthetic quality must be evaluated to provide feedback to the name-placement rule processor. If this quality is above the current threshold, the placement is acceptable. If not, another placement rule must be tried.

The quality measurement algorithms determine the acceptability of a name placement. Some of the algorithms determine whether a name overlaps existing names or features or if the name is curved excessively, making it difficult to read. The quality measures also determine if a feature has been labeled or should be labeled. Others determine if a name is off the map or outside area feature boundaries. Some measures indicate a placement is poor if it is partially inside a region and partially outside. The algorithms are also used, for example, to prevent a route number from being placed over the intersection of two highways.

When evaluating the quality of a name placement, each character in the name must be examined to determine whether it overlaps other characters or symbols. Character and symbol locations, defined in terms of longitudes and latitudes, are placed into k-d trees. A region query is then performed to determine if a character or symbol already occupies the same location.

Performance

The time required to place names on two maps of similar density should be proportional to the number of names on the map, since the average number of rules used to place each name on these two maps should be similar. As the map becomes denser, more name-placement algorithms must be tried for each name. Therefore, the time required to place names will increase somewhat faster than the number of names on the map.

IMPLEMENTATION and RESULTS

The name-placement system described here was implemented on two different computer systems, a Prime 750 minicomputer and a Cromemco System Three microcomputer. A Hewlett Packard 7550 pen plotter was used to generate hard copies of the maps.

The 700 fully documented Fortran modules of the name-placement system were designed to be portable. All software was also designed using data abstraction and information hiding techniques.

Two maps were produced to illustrate the capabilities of the automatic name-placement system. The map in Fig. 6 is a 1:26,000 scale street map of Troy, New York. The map in Fig. 7 is a 1:1,160,000 scale regional map of Central New York State.

The Troy map illustrates dense-map name placement when most name-placement possibilities are very limited. Over 2,000 characters were placed on the map to label half of the 400 features. Streets are represented by parallel lines with their names or route numbers

in the center. In dense areas, these names may extend beyond the end of the street. Town names are centered, whereas county names are placed along borders.

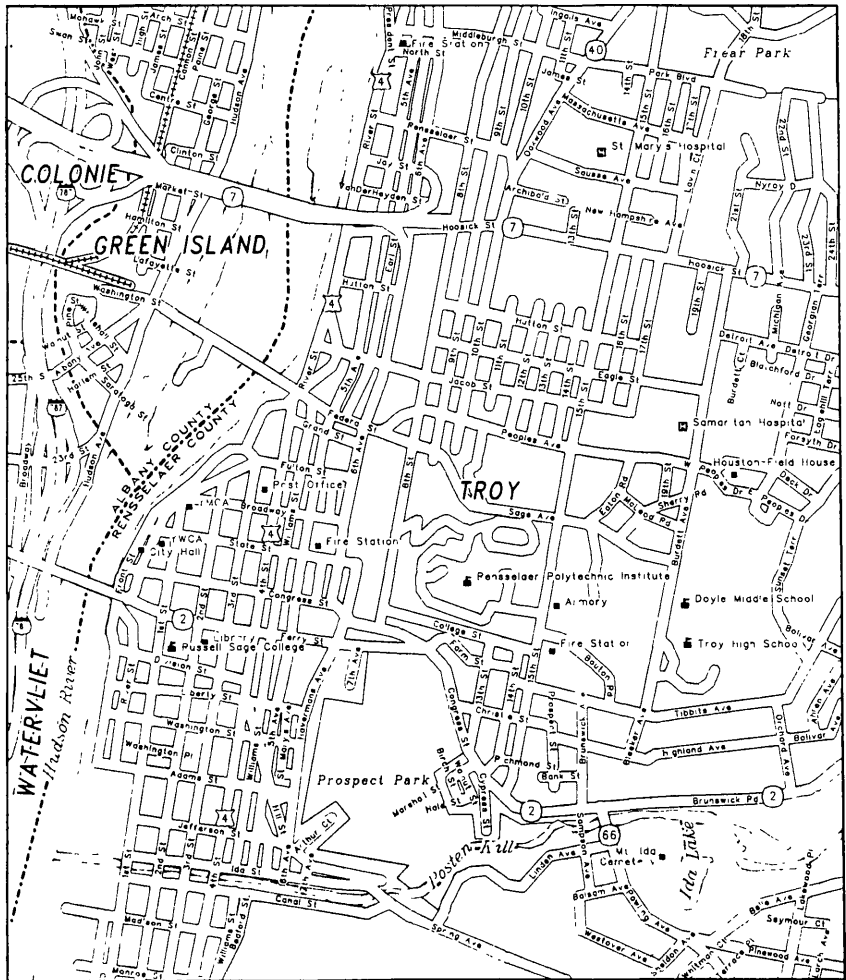
The regional map of central New York State shown in Fig. 7 has over 3,800 features consisting of 3,230 lines and 1,770 points. The lines are formed from over 28,000 individual segments. Approximately 18,000 characters have been placed on the map. This map is denser than the Troy map; however, there is more freedom to place names. This map clearly illustrates the capabilities and limitations of the dense-map name-placement system.

CONCLUSION

A rule-based system was developed to automate map production. The overall system creates map databases, places names, and plots maps of high aesthetic quality. All map data used by the name-placement system is based on features, not the graphics representing the features. This approach allows greater flexibility to define map styles and place names. If feature information were not present, it would be almost impossible to label a dense map adequately.

REFERENCES

- Ahn, J. and H. Freeman. 1983, A program for automatic name placement: In *Proceedings Auto-Carto Six*, pages 444-453, Ottawa, Canada.
- Basoglu, U. 1982, A new approach to automated name placement: In *Proceedings Auto-Carto V*, pages 103-112, Crystal City, Virginia.
- Bentley, J. L. 1975, Multidimensional binary search trees used for associative searching: *Communications of the ACM*, 18(9):509-517.
- Doerschler, J. S. 1987, *A Rule-Based System for Dense-Map Name Placement*. Technical Report SR-005, CAIP Center, Rutgers, P.O. Box 1390, Piscataway, New Jersey 08855-1390.
- Freeman, H. and J. Ahn. 1984, Autonap - an expert system for automatic map name placement: In *Proceedings International Symposium on Spatial Data Handling*, Zurich, Switzerland.
- Hirsch, S. A. 1982, An algorithm for automatic name placement around point data: *The American Cartographer*, 9(1):5-17.
- Imhof, E. 1975, Positioning names on maps: *The American Cartographer*, 2(2):128-144.
- Kelly, P. C. 1980, *Automated Positioning of Feature Names on Maps*. Master's thesis, Department of Geography, State University of New York at Buffalo, Buffalo, New York.
- Langran, G. E. and T. K. Poiker. 1986, Integration of name selection and name placement: In *Proceedings Second International Symposium on Spatial Data Handling*, pages 50-64, Seattle, Washington.
- Merrill, R. D. 1973, Representation of contours and regions for efficient computer search: *Communications of the ACM*, 16(2):69-82.
- Yoeli, P. 1972, The logic of automated map lettering: *The Cartographic Journal*, 9(2):99-108.



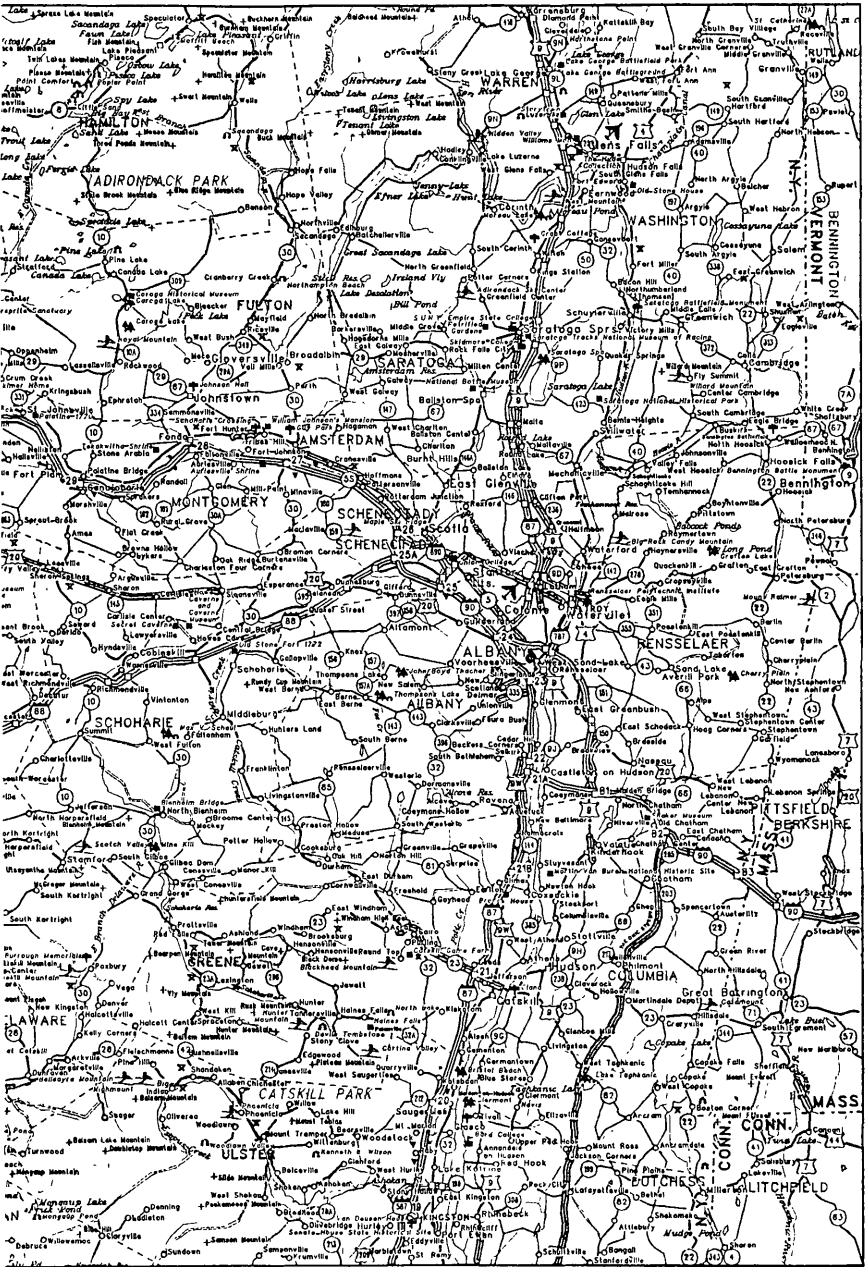


Fig. 7: Example of Automatic Name Placement (State Map)

THE USE OF ARTIFICIAL INTELLIGENCE IN THE AUTOMATED PLACEMENT OF CARTOGRAPHIC NAMES

David S. Johnson
Dr. Umit Basoglu
Intergraph Corporation
One Madison Industrial Park
Huntsville, AL 35807-4201

ABSTRACT

The placement priorities of cartographic names differ from one mapping agency to another. They also differ from one product to another within a single agency. Automated name placement software needs to be flexible enough to handle varying needs. Rule based processing provides this flexibility. Using product rules, placement is achieved for point, line and area features and conflicts pertaining to text are detected and resolved. The development of the product rules can be a very time-consuming process. The time necessary to develop these rules can be reduced through the use of neural network technology.

INTRODUCTION

In the past, the placement of cartographic names has been a manual process which has generally consumed up to fifty percent of the map preparation time (Yoeli, 1972). More recently, cartographers have sought the help of a computer to reduce the time required to annotate maps. Many different algorithms have been developed to aid in this process. Among them are the priorities suggested by Yoeli (1972) for the placement of names for point features, the wavefront approach used for the placement of names for area features (Montanari, 1969), and the placement for linear names (Basoglu, 1984). While these are some of the more recent algorithms, they do not meet all of the needs of the cartographer.

The priorities suggested by Yoeli specify where the optimum text placement location is (e.g. upper-right), and the order that other locations should be tried when conflicts prevent the placement of the text. Yoeli's priorities work well when used for the placement of names associated with populated places. However, other cartographers prefer to use a different set of rules for that task. In addition, there is issue of what rules should be used for other types of point features. For example, the rules for the placement of elevation values associated with control points usually differ from the rules for the placement of names associated with populated places. Similarly, there are many alternative ways for labeling various area features and linear features.

The rules for labeling a feature depend on both the text placement philosophy of the cartographer who is positioning the text and the type of feature being processed. This means that any system designed to automate this process must be robust enough to support several placement algorithms and flexible enough to allow the cartographer to dictate when a particular algorithm should be used. The system should also be expandable to support the addition of new placement algorithms. Finally, the system should be intelligent enough to aid the cartographer in generating the placement rules based on existing maps. Through the use of artificial intelligence techniques, such a system is feasible.

According to Klimasauskas (1988), the major information processing techniques are "procedural languages (BASIC, FORTRAN, COBOL, C), expert systems, and neural networks". A combination of all three is needed to develop a system that meets the requirements stated above.

THE USE OF PRODUCT RULES

An expert system is a system with a high degree of skill or knowledge about a certain subject. Two important roles in the development of an expert system are the role of the knowledge engineer (the person who implements the system) and the role of the domain expert (the person who is an expert in the domain of the task to be performed by the expert system). To implement an expert system, a knowledge engineer must be knowledgeable about the current techniques being used by the domain expert. Knowledge engineers gain their knowledge by having direct consultations with domain experts, by reading material written by domain experts, and/or by studying the results produced by domain experts.

The knowledge engineer and the domain expert often work together to come up with a list of the basic functionalities needed to perform the desired task. After the basic functionalities have been defined, it is the knowledge engineer's job to write the software that supports those functionalities. The domain expert writes or assists in writing a set of rules that specify when a particular functionality is to be used. For example, some information for en route airways on navigational charts is to be placed along the airway at a specified distance from the airport while other information is to be placed near the midpoint of the airway. The basic functionality needed to support this is placing text along a linear feature at a specified distance from a given endpoint of that feature and placing text at the midpoint of a linear feature. The airways example illustrates two functionalities commonly needed when processing linear features. If the system is to be used for more than labeling airways, then many more different functionalities will be needed.

The knowledge engineer and the domain expert should identify and incorporate as much functionality into the system as feasibly possible when the system is being designed. Chances are, as time goes by, more functionalities will be requested by the users of the system. In addition, some of the original algorithms may become outdated. Because of this, the system needs to be able to support the addition of new algorithms and the revision and replacement of existing algorithms. This is easily handled by the use of rulesets.

A simple example of what a rule might look like is as follows:

```
If a feature has a feature_code of "xx...x" and
    has a population greater than "50000"
then
    send load_text to the "text_processor" using city_name
    send compute_point_text_location to the "text_processor" using "GT_50000"
    send place_text to the "text_processor" using feature
```

The "if" clause ensures that only cities with populations greater than 50,000 will be processed by the then clause of this rule. The "then" clause loads the city name, computes the position of the text based on a parameter code (GT_50000), and then places the text, linking it with the point feature. The parameter code is used by the compute_point_text_location algorithm as a look-up into a table of parameters containing information about point features. This information contains, among other things, the symbology and the optimum placement location to be used when placing the text, optional placement locations to be used in case a conflict occurs when

attempting to place the text in the optimal location, and lists of what features and colors are to be considered as conflict. The algorithm will never place text on top of existing text.

If the `compute_point_text_location` algorithm is enhanced, the ruleset will remain unchanged unless the arguments to the algorithm change (in the example above, the parameter code is the argument to the algorithm). If the user decides to use a different algorithm for labeling cities with a population greater than 50000, then two things must be done:

- 1) the new algorithm must be written and added to the existing set of functionalities
- 2) the ruleset must be modified to invoke the new algorithm instead of the current algorithm.

Much progress has been made in defining the necessary functionalities for placing map text and in developing computer algorithms which implement these functionalities (Freeman and Ahn, 1984). Although mapping agencies may want to implement their own placement algorithms, the algorithms suggested by Freeman and Ahn provide an excellent start for the implementation of an expert system.

Once a functionality has been added, it can be used for many different types of features. The domain expert can mix and match functionalities as needed, without affecting the underlying code. If a mapping agency decides to add a different type of map to its product line, then most, if not all, of the needed functionality will already exist. All that will be necessary is the creation of a ruleset specific to that type of map.

THE USE OF NEURAL NETWORK TECHNOLOGY

One area of artificial intelligence that is becoming more and more popular is neural networks. In the past, some people have viewed neural networks as a competing technology of expert systems. However, these two technologies can be combined to complement each other. Conceivably, neural networks could be used to assist the domain expert in the generation of product rulesets.

Neural networks are good at examining existing data and forming assumptions based on that data. They accomplish this by means of assigning weighted values to each functionality. Some experts say the human brain operates in much the same way. The following is an excerpt from Klimasauskas(1988) explanation of neural networks: "The basic information-processing unit in neural networks is the processing element, or neuron. These terms are used interchangeably. Brain researchers have identified over 100 different kinds of neurons. Processing elements also come in a variety of types. Each type, either in artificial or biological neural networks, operates in a specific way which assists in implementing a specific function. Common to all of them is that one or more inputs are modulated by connection weights to change the stimulation level internal to the neuron. Based on this internal stimulation or activation level, the neuron may or may not produce an output. The output is related to the internal activation level, but this relationship may be a non-linear or discontinuous function".

This same principle can be applied when devising a method of assisting the cartographer with the generation of product rules based on existing maps. By examining an existing placement location and the conditions surrounding the choice of that placement location, assessing the probability that the location is the optimum location, and storing this probability as a weight in the neural network, it is possible to "learn" the rules used by the cartographer when placing the text.

For example, if a point feature has text associated with it, the area surrounding the point can be examined to infer why the point was labeled in the manner that it was labeled. A simple case would be an oil well symbol with no other features close enough to the symbol to cause a conflict with any of the other point text placement positions. If there are no potential conflicts in the proximity of the oil well, then it may be inferred that the placement location used to label the oil well represents the optimum placement for text associated with oil wells. This is because the cartographer was not limited by any conflicts when deciding where to place the text. Since the cartographer could have chosen any of the placement locations, it can be inferred that the text was placed at the optimum placement location.

It is assumed that the functionality of the point placement algorithm includes ten different locations for the placement of point text. The goal of the neural network is to examine the map and to assign a weight to each of the ten locations. Based on the resulting weights, the rules and parameters used to label the point features can be calculated. The next step is to devise a means of computing the weight of each location. All weights should initially be set to zero. As each point is processed, the text placement location should be determined, a weight for that location should be computed, and that weight should be used to compute an average weight for that point text placement location. The weight should be computed as the number of possible placement locations minus the number of locations which could not be considered because of conflicts.

A weight of ten would be assigned to the location described in the example above because there were no conflicts. If there had been potential conflicts in three of the alternative locations, then a weight of seven would have been assigned to the location used to place the text. This newly computed weight would be used in the computation of an average weight for that location. That is,

$$\text{avg}(n+1) = \frac{\text{avg}(n) * n + \text{wt}(n+1)}{n + 1}$$

where avg is the current average weight for a given location, n is the total number of weights that have been used to calculate the average weight, and wt(n+1) is the newly computed weight of the feature being processed. The more a well features the neural network can examine, the more accurate its weights will become. The location that ends up with the highest weight will be considered the optimum placement location for oil well symbols.

EXTENSION OF NEURAL NETWORKS FOR CARTOGRAPHY

The above example illustrates how the traditional implementation of neural networks could be used to determine the optimum location for label placement for point symbols. The discussion below suggests an enhancement to the traditional implementation to speed up the learning process and to improve on the calculations of the weights.

It could be that for ninety percent of the cases where the text was placed in the cartographer's optimum location, there were at least two conflicts in other locations. Although those conflicts did not affect the choice of placement location, they will affect the calculations. This situation could result in the average weight for the cartographer's optimum placement location being equal to approximately eight. It is possible that for ninety percent of the cases where the text was placed in the cartographer's second-best location, there were conflicts in only one other location (which would have to have been the optimum location). This situation could result in the average weight for the cartographer's second-best placement location being equal

to approximately nine, which is higher than the weight that was assigned to the cartographer's optimum placement position. Ideally, this situation would correct itself as more and more maps are examined. To avoid the above problem, the learning process can be helped by assigning a "locked-in" factor to each of the weights.

Initially, none of the weights would be considered locked in. Consider the first example where a feature exists with no conflict; it is known for certain that the placement position is the optimum position. When the weight that is to be assigned to a placement position is known for certain, that position can be marked as locked in. Once the optimum placement location has been locked in, it is possible to lock in the second-best location, then the third-best location. The second-best location could be locked in when text is placed in that location and there is only one other location containing text. Since that one other location has been locked in as the best location, it can be inferred that this location is the second best location. The process would continue for remaining locations.

The user should be able to tell the neural network what to do if text is encountered in a location not represented by the ten given locations. For example, the neural network could be told to flag any such text so that a new placement location can be added to the list of existing positions, or it could be told to ignore the text. The neural network should also be knowledgeable enough to recognize some of the special cases. For example, when placing text associated with a point feature that is located on a land mass and near a water body, some cartographers prefer to place the text so that it is located entirely within the land mass or within the water body.

Text locations for area features and linear features can be "learned" in a fashion similar to the way text locations for point features can be learned. But so far, the example has been oversimplified. Referring back to the sample rule, it can be seen that text location is only one aspect of the rule. Other important information used by the sample rule is the population, the city_name, and the feature_code that was used as an index into a table of parameters. Creating a neural network capable of generating a rule similar to the sample rule for each type of feature would be difficult, impractical, and in some cases impossible. However, if a mapping agency likes the appearance of a particular set of maps and wants to learn more about the rules used to generate these maps, then the use of neural network technology will be useful. Methodology to find all point features, to divide them according to feature code, and then to determine the point placement priorities used when placing the text for each type of feature are possible enhancements. This information could be used simply to generate a report or the software could use the information to generate a shell of a ruleset that would later be completed by the cartographer. The software could also examine the different symbologies used to represent the point features and use that information to generate a shell of a parameter file that could later be completed by the cartographer. Information about area features and linear features could be processed similarly to the way point features are processed.

CONCLUSION

Manual placement of cartographic names is a time consuming process. Early research into automating this process was constrained by the software and hardware capabilities of that time. The average user now has access to more memory, more disk storage, and faster computers. Software technology has also continued to evolve. Research in the area of artificial intelligence has resulted in the development of many useful applications that are based on this technology. The technology of

expert systems can be used to automate the text placement process by using a system that is flexible enough to meet the needs of different users and that can grow with the user. The technology of neural networks can be used to complement the expert system by assisting in the creation of the product rules that are used by the expert system. The use of a "locked-in" factor associated with the weights in the neural network can be used to speed up the learning process. This factor is also useful when trying to determine how accurate the weights are. The larger the percentage of weights that are locked in, the more accurate the weights are likely to be.

REFERENCES

Basoglu, U., "A New Approach to Automated Name Placement Systems," Ph.D. Dissertation, University of Wisconsin at Madison, Madison, Wisconsin, May 1984.

Freeman, H. and J. Ahn, "Autonap - an Expert System for Automatic Map Name Placement", Proceedings International Symposium on Spatial Data Handling, Zurich, Switzerland, 1984.

Klimauskas, Casimir C., "Neural Networks: A Short Course From Theory to Application", PC AJ, Vol. 2, No. 4, November/December 1988, pp. 26-30.

Montanari, Ugo, "Continuous Skeletons from Digitized Images", Journal of the ACM, Vol. 16, No. 4, October 1969, pp. 534-549.

Yoeli, P., "The Logic of Automated Map Lettering", The Cartographic Journal, Vol. 9, No. 2, 1972.

RULE-BASED CARTOGRAPHIC NAME PLACEMENT WITH PROLOG

Christopher B. Jones
Anthony C. Cook
Polytechnic of Wales
Pontypridd
Mid Glamorgan
CF37 1DL, UK

ABSTRACT

Positioning text on maps is a complex task subject to numerous rules which may vary according to the purpose of the map and who is making it. Ideally therefore, automated systems for name placement should be flexible in terms of the rules used to control them. Because the logic programming language Prolog is essentially rule-based, it appears to be appropriate for developing such systems. In practice, it has proved possible to use the language for writing relatively short and clear programs to find non-conflicting positions for names and for specifying rules for selecting names and potential label positions. An experimental Prolog name placement system has been developed and has been used to place names on a variety of maps. The system provides access to a spatial database which facilitates the creation of rules which depend upon spatial relationships between names and other map features.

INTRODUCTION

The results of cartographic name placement can vary greatly according to the theme, purpose and style of the map onto which names are to be placed. In general, name placement may be regarded as subject to sets of rules which differ from one type of map to another, as well as between cartographers and cartographic organisations. The concept of rules in name placement was integral to Imhof's (1975) review of the subject and it has been incorporated in automated systems such as those of Freeman and Ahn (1984) and Pfefferkorn et al (1985). A major problem with implementing automated rule-based systems is to find a way of programming the rules such that they can be changed easily to meet user requirements. With this objective in mind, the logic programming language Prolog (e.g. Clocksin and Mellish, 1981) is of particular interest, as it uses rules and associated facts as its basic constructs, which are referred to as predicates.

In being a declarative, rule-based language, Prolog appears to offer potential for programming name placement at a relatively high level, provided that the cartographic rules can be translated into the syntax of Prolog. Thus the appeal of logic programming for name placement is the possibility of writing programs which consist of descriptions of rules rather than the algorithms for implementing them. Ideally these programs should be short, readable and easily amended. It is important to realise in this context that some of the apparent advantages of Prolog derive from the fact that the language uses a built-in inference mechanism to deduce solutions which obey the facts and rules which specify the problem. This may be contrasted

with conventional procedural languages (such as Pascal, Fortran and C) which work by executing a set of instructions based on an algorithm which specifies how to solve a problem.

Name placement can be thought of as a combination of two processes, one of which is concerned with resolving conflicts between name positions (or labels), while the other is concerned with the selection of names and their associated label characteristics and positions. The remainder of this paper falls into two parts, corresponding to these two processes. In the first part a logic programming strategy is described for finding combinations of name positions which minimise conflict between names and between names and unrelated features. This is confined here to the problem of labelling point-referenced features and is based on Jones (1987). The second part relates to the problems of selecting names and of generating trial or candidate labels which satisfy rules of text configuration and of graphic association between the text and the named feature. Examples are given for name placement problems which include point, line and area-referenced names. They are based on recent research by one of the authors who has developed a name placement system which uses Prolog for conflict resolution and for implementing rules for selecting names and appropriate labels (Cook, 1988). This system, called NAMEX, uses a spatial database which may be accessed from Prolog by means of calls to subroutines which are written in Fortran.

PLACEMENT STRATEGIES FOR AVOIDING CONFLICT

In order to find a set of name positions which satisfies rules of conflict avoidance with other names and features, it is necessary to provide mechanisms for generating trial positions for each name to be placed and for detecting whether any given position results in conflict. Let us assume that there is a Prolog predicate, `find_trial_position`, which on being called returns an horizontal trial position for a label (Name) defined by location co-ordinates X, Y in a raster image co-ordinate system and by its Length, which is measured in pixels. This predicate could take the following form:

```
find_trial_position(Name, X, Y, Length).
```

The presence of horizontal labels which have been placed can be recorded with predicates of the form

```
label_at(Name, X, Y, Len).
```

where X and Y are the co-ordinates of the start of a row of pixels of length Len occupied by the label. Depending on the height of the text, several such predicates could be asserted for each label. For simplicity of explanation, we assume here that only a single row is required. To ensure that point symbols can be uniquely identified, so that a label's own symbol can be distinguished from others, the presence of symbols can be recorded in terms of rows of pixels defined by predicates of the form

```
symbol_at(Name, X, Y, Len).
```

where the parameters are as before. The presence of all other map features can be recorded in the Prolog database using additional fact predicates. One possible method is to encode map features in a run length format, as described in Jones (1987). Overlap or conflict detection can now be performed by a second predicate (no conflict) which is proven to be true if no conflict is found between the given position and any other label and, if necessary, any other map feature. It can do this by examining the positions of labels already placed and, if appropriate, by searching the database to identify the presence of other map features which could be affected by this possible placement. It may be defined in terms of a rule which first attempts to prove that there is an overlap, in which case it fails, otherwise it is true. For the purpose of testing for overlap with other labels, this can be expressed as follows (note that the symbols /* and */ are used for delimiting comments, while :- means 'if' and , and ; mean 'and' and 'or' respectively):

```
no_conflict(Xs, Ys, Search_length, Ignore):-
/*_ look for names or symbols on given scan line */
  (label_at(Name, X, Ys, Length) ;
   symbol_at(Name, X, Ys, Length)),
  not(Name=Ignore), /* should name or symbol be ignored */
                        /* now test for overlap */
  Search_end is (Xs+Search_length-1), Search_end >= X,
  Data_end is (X+Length-1), Xs <= Data_end,
  !, fail. /* fail completely if any overlap found */
/*_ succeed only if not possible to prove overlap */
no_conflict(,_,_,_).
```

This definition can also be used in situations where all map features must be considered, provided that the trial positions given by the predicate find_trial_position have already taken account of them. Such an approach has the merit of performing all spatial data searches, for potential feature conflicts, once only in a preprocessing stage, rather than in the course of conflict resolution. Alternatively, as described in Jones (1987), the logic program database may be loaded with a complete description of the map, which can be searched when the no_conflict predicate is called. This may however introduce repetition of searches.

A relatively simple strategy can now be applied in which, for each name, a predicate place_label uses the above predicates to generate a possible position which is tested for overlap. If the position is found to be acceptable, it is recorded in the logic program database by means of a predicate record_position which asserts label at predicates. When placing several names however, it may be impossible to place an individual name because other previously placed names are in conflict with it. Such a situation induces backtracking in which one or more previously placed labels will be removed and alternative trial positions used, before going forward to try again with the label which could not be placed. To enable this backtracking, the predicate for placing a label calls a predicate (clear_previous_label) which retracts from the Prolog database any previously asserted label at predicates for that name, before any other is recorded. The place_label predicate may be described as follows:

```

place_label(Name):-
    find_trial_position(Name, X, Y, L),
    no_conflict(Name, X, Y, L),
    clear_previous_label(Name),
    record_position(Name, X, Y, L).

```

Placement of a group of labels can be done by applying the `place_label` predicate to all labels to be placed. One way of handling the group of labels is to put them in a list. Lists in Prolog can be referred to in terms of the first item in the list (Head) and the remainder of the list (Tail). When a list is handled in this way, it is symbolised by `[Head|Tail]` where the square brackets represent the boundaries of a list. Thus a predicate `place_group`, which attempts to place all names in the group, can be described in the following manner:

```

place_group([]).      /* terminating condition (empty list) */
place_group([Head_name|Tail]):-
    place_label(Head_name),    /* place first name in list */
    place_group(Tail).        /* place remaining names */

```

This predicate is defined recursively, such that when the first label has been taken from the head of the list of labels, it is passed to the predicate which attempts to place it (`place_label`) before passing on the list of remaining names to the original predicate (`place_group`). If `place_label` cannot find a suitable position, and hence fails, the `place_group` predicate backtracks to the last situation in which an alternative action was possible. Such a possibility would occur when another trial position was available for a previous name. Note that if a previous label cannot find another position, it will induce further backtracking to its predecessor in the list of names. The logic program will fail only when all combinations have been exhausted without success. Such complete failure can however be averted by deleting a name (such as the last one which failed to be placed, if all names have equal importance).

Reducing Search Time

When there are many names to be placed, program run times may be unacceptably long. Solution times can however be reduced by breaking the problem into sub-problems defined by groups of names which are in potential overlap with each other. This was done by Freeman and Ahn (1984), who referred to these groups as connected components. Further reduction in search time can be obtained by sorting trial label positions in order of decreasing difficulty, as also found by Freeman and Ahn. Possible measures of difficulty for a particular name include the number of trial positions available and the number of neighbouring names which are in potential overlap.

RULE-BASED SELECTION OF TRIAL LABELS

The techniques for conflict avoidance, as described in the previous sections, are based on the assumption that a mechanism exists whereby trial label positions are generated, either in advance or in the course of seeking non-conflicting positions. The process of selecting trial positions is one which has a major bearing on the style and,

in many cases, the legibility of the resulting map. Trial positions are selected according to factors such as orientation, the relationship between the label and the feature which it annotates, and its relationship to other map features. The relationship between a label and its associated feature depends upon the class of feature concerned. For points, the label is typically placed immediately adjacent, to the right or left, above or below or at intermediate positions to these. Line labels might be placed on the line or to one side or the other of the feature, or perhaps lie across it. Area labels may be positioned inside a region at some orientation which could be related to the shape of the area, or they might be outside, adjacent to the edge, such as when the area is too small to accommodate the label.

The combination of factors which determine trial positions, along with issues of whether or not particular features should be labelled at all, provide enormous scope for controlling the design of a map. In an automated system, these factors should therefore be subject to rules which can be adjusted by the cartographer. In the NAMEX name placement system (Cook, 1988), an attempt has been made to demonstrate how, with the aid of logic programming, such rules can be implemented in a manner which allows them to be changed relatively easily. In the present implementation, such changes still depend in many cases upon a knowledge of Prolog programming but, in the ideal situation, it is possible to envisage a user interface which allowed rules to be changed via a natural language dialogue.

Before looking at examples of Prolog rules in NAMEX, it should be noted that the system is a hybrid one in terms of programming language, in that a number of low level functions, concerned with tasks such as accessing a spatial database and detecting overlap, have been implemented in Fortran using subroutines which can be called from Prolog. Of particular significance for the implementation of spatially defined rules are functions which determine the presence of specified map features within a given vicinity (e.g. a circle or rectangle). It should also be remarked that the NAMEX conflict resolution strategy differs somewhat from that which was described in the previous section.

Label Selection in NAMEX

Rules for generating trial labels in NAMEX fall into three categories. The first is concerned with the selection of names for which label positions are sought, the second defines label configuration, which relates to text size, orientation and position relative to the named feature, while the third is concerned with validation of the selected configurations. Validation ensures that selected configurations obey rules of association between labels and adjacent features. Thus there may, for example, be priorities and limits controlling what features may be overlapped, and what distances there may be between labels and nearby features.

The extent to which rules must be provided for the initial selection of names depends upon the application. In some situations, it may be desirable to attempt to plot all names, in which case selection may depend upon either an

initial analysis of the map space and name density (as in Langran and Poiker, 1986), or perhaps on a system of priorities or ranks which must be considered when the conflict resolution strategy encounters difficulties (at present, the NAMEX conflict resolution strategy can delete names irrespective of their cultural importance). There are also situations in which it is appropriate to specify rules which govern selection of features to be named according to the theme of the map and on the basis on map scale.

At a very simple level of selecting major feature types, a predicate name_select(Fsn,Ftype) has been used, in which Fsn is a unique feature serial number, and Ftype is the feature type (point, line, area). In order to ensure that, say, all point features are to be selected, the predicate name_select(,point) may be asserted. If the feature name under consideration has the corresponding value for Ftype then, when the predicate is called for the given name, it will succeed (i.e. be regarded as true). More sophisticated rules which take account of scale have been used in the creation of a series of Moon maps in which craters are labelled if their size exceeds a threshold which is determined by an empirical function dependent upon map scale. For example, a predicate for selecting primary craters on maps of scale smaller than 1:15,000,000 may be defined as follows:

```
select_crater(primary, Scale, Fsn):-
    Scale > 15000000,
    crater_diameter(Diameter, Fsn),
    generalise_crater_rule(Limit, Scale),
    Diameter >= Limit.
```

The predicate crater_diameter accesses the NAMEX database to determine the diameter of feature Fsn, while the generalise_crater_rule predicate calculates the diameter threshold Limit, on being given a value of Scale. The map in Figure 1 was created using such a generalisation rule.

Many of the rules of configuration selection may be implemented quite simply, since there may often be a direct and fixed relationship between feature type and the label's size, orientation and position relative to the feature. For example, control over the orientation of labels can be achieved with a predicate select_label_orientation(Fsn,Ftype,Fcode,Orientation), in which Fcode specifies a particular class of feature of specified type, and Orientation may be either horizontal or diagonal. Once a diagonal orientation has been selected, the angle employed is calculated as a function of the disposition of the feature. Normally the first three parameters would be predetermined for a given label. Calling the predicate will then result in Orientation being instantiated with an appropriate value. In the case of a rule which specified that all point-referenced labels were to be plotted horizontally, it would be sufficient to use the predicate

```
select_label_orientation(Fsn,point,Fcode,horizontal).
```

Thus, on calling the predicate, all point features will be given a horizontal orientation. The orientation of line labels may often be subject to a variety of rules. If it

was required to set labels to be horizontal for all roads with feature code 'a_road', it could be done with the predicate

```
select_label_orientation(Fsn,line,a_road,horizontal).
```

If however the orientation of 'a_road' labels was to be a function of direction, the predicate would have to access the database to examine the properties of the specified feature. The following rules would set the orientation to be horizontal if the line was orientated more steeply than 40 degrees, and otherwise diagonally.

```
select_label_orientation(Fsn,line,Fcode,Orientation):-
    select_line_label_orientation(Fsn,Fcode,Orientation).
```

```
select_line_label_orientation(Fsn,a_road,Orientation):-
    compute_angle_of_line(Fsn,Angle),
    select_orient_by_angle(Angle,Orientation).
```

```
select_orient_by_angle(Angle,diagonal):-
    Angle =< 40.
```

```
select_orient_by_angle(Angle,horizontal):-
    Angle > 40.
```

Rules of this sort were used in placing major ('A') road names on the road map in Figure 2. Similar rules have been implemented for controlling the orientation of area labels, as in Figure 3. Here it was required to determine the label orientation according to the degree of elongation and the orientation of the area itself, both of which were found by calls to standard NAMEX predicates.

An important aspect of name placement is that of maintaining a clear association between the label and the feature it annotates. As an example, the predicate `valid_position`, described below, ensures that a point label does not overlap too many other features or lie too close to any adjacent point symbols, other than its own.

```
valid_position(Fsn,Ftype,Fcode,Pos_number):-
/* get location of label at given position number          */
    read_label_data(Fsn,Ftype,Fcode,Pos_number,
                    X,Y,Angle,Length,Height,Prox),
/* examine underlying pixels in rectangle centred on X,Y  */
    raster_rectangle_totals(X,Y,Angle,Length,Height,
                            Number_of_pixels>Total_pixel_value),
    Ratio is realof(100 * Total_pixel_value /
                    Number_of_pixels),
/* find current acceptable threshold value for ratio      */
    dense_space_threshold(Thresh_dense),
/* test ratio against threshold                            */
    Ratio =< Thresh_dense,
/* enlarge label region by Prox to include buffer zone    */
    enlarge_label(Length,Height,Prox,Newlength,Newheight),
/* test for absence of illegal features in label region  */
    raster_rectangle_features_absent(X,Y,Angle,
                                      Newlength,Newheight,[city,town,village]).
```

The predicates called by `valid_position` are not defined here. Some of them make calls to lower level predicates which access the contents of the NAMEX database. Note that

the total pixel value found by raster_rectangle totals is obtained from the sum of priority weighted raster bit planes representing map features, and that the test for illegal pixel values in the extended region of the label depends upon the pixels of the label's feature having been temporarily masked. The logic described here was used in creating Figure 2.

Comparable techniques to these can also be used to good effect in validating on the basis of other cartographic criteria which require spatial search. An example occurs in the county map (Figure 3), in which settlement names away from the coast have been placed either inside or mostly inside the county to which they belong. The rule was formulated with reference to searches in the rectangular regions formed by the co-ordinates of the named settlement and the centre of the label itself. Substituting the feature type 'county boundary' into the list of features to search for in the raster_rectangle features absent predicate, it was possible to test whether a county boundary lay between the label centre and its point symbol. If the county boundary was also a coastline, then the rule was not applied. To keep a label entirely in its own county, the search rectangle could be enlarged appropriately.

CONCLUSIONS

The logic programming language Prolog has been used for writing rule-based programs which perform cartographic name placement. The built-in inference mechanism of the language makes it possible to produce relatively short and readable programs which find positions for groups of names by obeying simple rules of conflict avoidance. Rules for the selection of labels to be placed, and the selection of suitable label configurations, have also been implemented in a name placement system (NAMEX) which provides access to a spatial database for the purpose of evaluating potential label positions. NAMEX has been used to place names on a variety of maps subject to quite different rules. Useful future developments of the system include more sophisticated conflict resolution strategies governed by rules controlling the deletion of names in crowded areas, and a high level user interface allowing rules to be encoded via a natural language dialogue.

ACKNOWLEDGEMENTS

The authors are grateful for financial and material assistance from the Ordnance Survey, Southampton. ACC was supported by an SERC studentship.

REFERENCES

- Imhof, E., 1975, "Positioning Names on Maps", The American Cartographer, Vol. 2, No. 2, pp. 128-144.
- Clocksins, W.F. and C.S. Mellish, 1981, Programming in Prolog, Springer-Verlag.
- Cook, A.C., 1988, "Automated Cartographic Name Placement Using Rule-Based Systems", PhD Thesis, Polytechnic of Wales.

Freeman, H. and J. Ahn, 1984, "AUTONAP - An Expert System for Automatic Map Name Placement", Proceedings International Symposium on Spatial Data Handling, Zurich, pp. 544-569.

Jones, C.B., 1987, "Cartographic Name Placement With Prolog", manuscript, Dept. Computer Studies, Polytechnic of Wales.

Langran, G.E. and T.K. Poiker, 1986, "Integration of Name Selection and Name Placement", Proceedings Second International Symposium on Spatial Data Handling, Seattle, pp. 50-64.

Pfefferkorn, C., D. Burr, D. Harrison, B. Heckman, C. Oresky and J. Rothermel, 1985, "ACES: A Cartographic Expert System", Proceedings Auto-Carto 7, Washington DC, pp. 399-407.

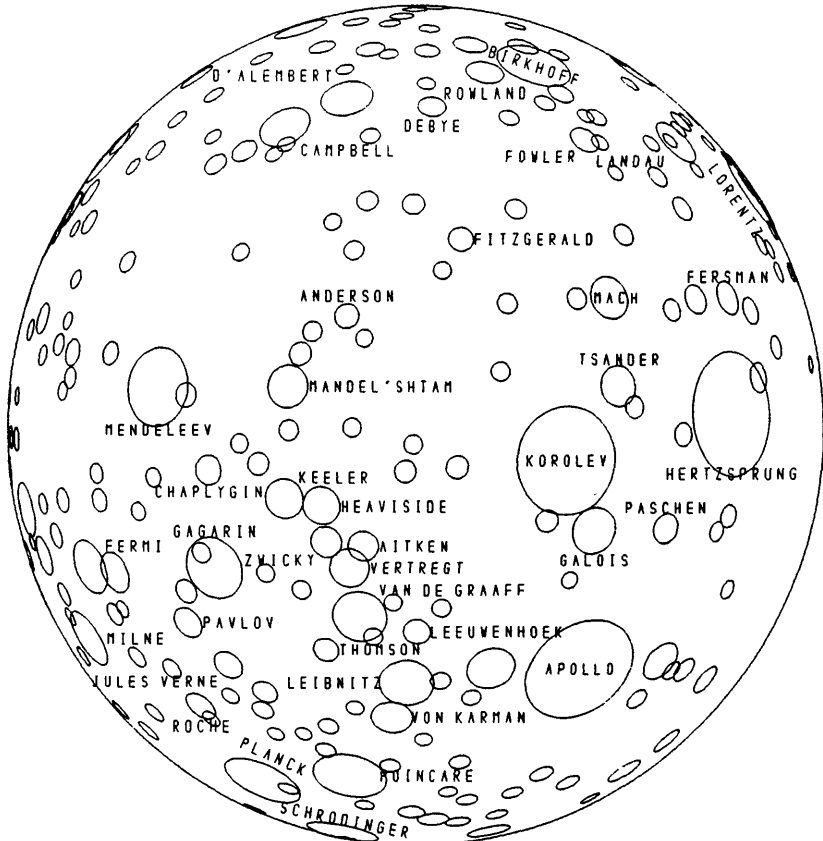


Figure 1. Far side of the Moon

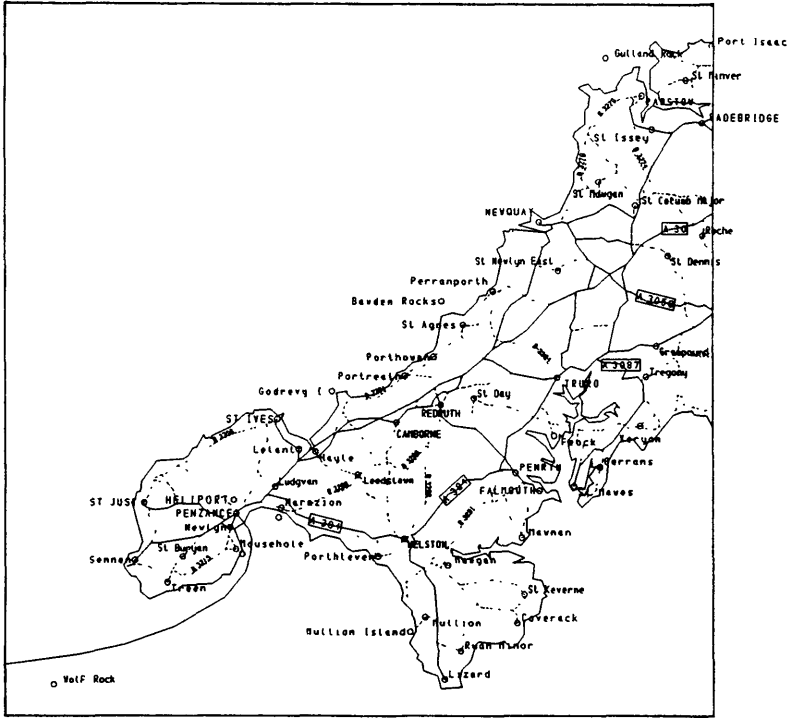


Figure 2. Road map (using Ordnance Survey data)

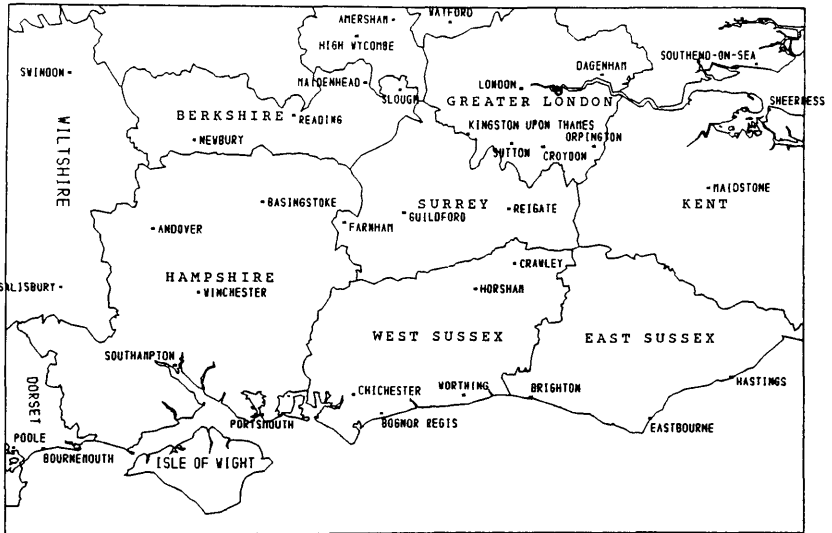


Figure 3. County map (using Ordnance Survey data)

THE DEVELOPMENT OF DIGITAL SLOPE-ASPECT DISPLAYS

A. Jon Kimerling
Department of Geography
Oregon State University
Corvallis, Oregon 97331

Harold Moellering
Department of Geography
Ohio State University
Columbus, Ohio 43210

ABSTRACT

Slope-aspect information is widely used by earth scientists, environmental planners, and other analysts dealing with this facet of the physical landscape. Digital elevation model data in raster form are commonly employed to compute for each pixel the aspect azimuth and aspect class within which the azimuth falls. Various coloring schemes for displaying slope-aspect classes have been tried in the past. While most schemes employ hue differences to create visually distinguishable map classes, little attention has been given to visually relating aspect to the underlying landform. In this paper coloring schemes that simulate relief shading while maximizing visual discrimination of individual classes are presented for four and eight class slope-aspect maps, and the theoretical basis for color selection is reviewed.

INTRODUCTION

Slope-aspect, the compass direction of the maximum slope at a particular location on a surface, is a landscape characteristic fundamental to building site analysis, solar access planning, watershed management, and a host of other scientific and management activities. Although determining the aspect at a single location may be sufficient for some problems, most require an understanding of the pattern of slope-aspect variation across the landscape. Slope-aspect maps provide this regional view and are a required product in many instances. Computer produced slope-aspect maps are created most frequently, since aspect computation based upon grid cells in a digital elevation model (DEM) is a straightforward and efficient procedure as long as the elevations of adjacent cells can be retrieved quickly. Creating a visually effective display of the computed aspect azimuths on color CRT's or hardcopy devices is still a cartographic challenge, even when precisely calculated azimuths are generalized into a limited number of classes such as the four or eight cardinal compass directions. The foremost map design problem is to develop a cell coloring scheme that will maximize color contrast among classes while allowing the user to visualize the underlying landscape. In this paper past efforts are reviewed and new solutions to this long-standing map design problem are offered.

PAST ATTEMPTS

Numerous coloring schemes have been used on published slope-aspect maps. Although sequences of gray tones and sets of areal patterns have been used occasionally, coloring by hue differences predominates. The spectral hues of red, orange, yellow, green, blue and violet, and mixtures thereof, produce on some maps easily distinguishable classes in the map legend and within the map. These hues at varying levels of value and chroma have been either randomly assigned to classes or organized into full or partial spectral progressions. Yellow and neighboring lighter hues have been used to represent virtually all aspect directions, with an apparent association between "solar" yellow and south facing slopes found on many maps. Attempts to combine aspect and landform information have been restricted to overprinting aspect colors on a relief shaded base map, with generally disappointing results since shading inherently decreases our ability to distinguish among classes. Past attempts clearly indicate that slope-aspect display on maps needs a firmer theoretical footing.

SLOPE-ASPECT DISPLAY THEORY

Slope-aspect is a nominal level phenomenon since a particular aspect angle (azimuth or compass point) cannot be thought of as lesser or greater in physical magnitude or rank than any other. Hue and pattern differences are the correct visual variables to use when graphically portraying nominal level area phenomena, with hue differences used more often in computer mapping and geographic information systems. The set of hues selected for slope-aspect must allow the map reader to easily distinguish among classes, and yet see that aspect classes form a circular progression where adjacency implies greater inherent similarity. This rules out the use of randomly selected hues and points to the use of opponent-process colors.

Opponent process color theory "is a model of human color perception that predicts that there are four unique hues, with all others appearing as mixtures" (Eastman 1986). The opponent process model of human vision is based on the idea that although the cone cells in our eyes are sensitive to blue, green or red light, the ganglion cells linking the cones to the optic nerve interact to produce four perceptually unique colors -- red, blue, green and yellow. All other hues will be seen as mixtures of these "pole" colors, except that yellowish blues and reddish greens are not possible. These "pole" colors form the set of maximally different hues and hence are the easiest hues to distinguish. The implication for slope-aspect mapping is that a four class map should be symbolized with these "pole" hues if class discrimination is of paramount concern, whereas an eight class map should employ a progression of these hues alternating with the "mixture" colors of purple, blue-green, yellow-green and orange. These eight hues will be seen as a circular progression of related colors with mixture hues inherently similar to their two "poles".

Opponent process theory guides the selection of hues that are maximally discriminable yet seen as forming a circular progression of related colors, but the problem of hue assignment to particular slope-aspect classes remains. Assignment of "pole" hues to the cardinal aspect directions has been tried, but the resulting four class maps display the underlying land surface poorly. In some cases landforms are inverted and ridges appear to be valleys, and vice versa. Communication of slope-aspect information is enhanced when aspect class colors depict landforms correctly in a manner similar to relief shading where on north oriented maps northwest facing slopes are lightened and southwest slopes are darkened. The most fundamental form of analytical relief shading assumes an ideal diffusing surface with an apparent brightness that is proportional to the cosine of the angle formed between vectors representing incident rays coming from the northwest (315 degrees) and the surface normal (Horn 1982). Since slope-aspect alone is being mapped, the slope angle at each location is immaterial and can be assumed to be constant throughout. In this case the surface normal varies only with changes in aspect, and the cosine function describes the theoretical reduction in surface brightness that occurs as the aspect angle deviates progressively from the northwest incident illumination. Surface brightness can be thought of as proportional to light emitted from phosphors on CRT screens, meaning that the cosine law can be extended to electronic map displays.

Opponent process theory and the cosine shading law must be used in unison to create optimal slope-aspect coloring schemes for CRT displays. This translates to using yellow, the "pole" hue of highest inherent brightness and lightness, to display the aspect class centered on 315 degrees. Similarly, blue, the opposite "pole" hue of lowest lightness, is best applied to the class centered on 135 degrees. Red and green can be used interchangeably for the aspect classes centered on 45 and 225 degrees. An eight category map with red used for the 45 degree class would use these four hues plus orange, purple, blue-green, and yellow-green for the 0,90,180, and 270 degree classes, respectively.

Precise selection of the above "pole" and midpoint "mixture" hues, all with brightnesses close to falling on a cosine curve centered on yellow and scaled so that yellow equals one and blue zero, is somewhat subjective. However, the authors have discovered that, for the HLS color specification system used with Tektronix 4120 series terminals, holding all hues at maximum saturation (100) and progressively decreasing lightness from 50 for yellow to 40 for blue produces a visually effective hue progression that roughly follows the cosine lightness curve. Magenta and cyan, the true midpoint hues between the blue-red and blue-green "poles", were not found to work well regardless of lightness, since both are inherently much brighter than their "pole" hues and cannot be darkened so as to fall on the cosine curve without being "muddied" unacceptably.

CONCLUSION

Slope-aspect maps colored according to the above guidelines appear to be relief shaded as well, with landforms portrayed correctly and standing out clearly. The correct perception of landforms appears to enhance aspect recognition, since, for example, northwest aspects are seen as falling on northwest trending hillsides. This synergistic effect is seen on standard planimetric as well as on 3D-perspective slope-aspect maps displayed in either single image or stereoscopic image mode on recently introduced terminals such as the Tektronix 4126. Such judicious application of color theory should greatly improve the appearance and readability of future slope aspect maps.

ACKNOWLEDGEMENTS

This research has been performed as part of Professor Moellering's Spatial Data Display project funded by the NASA Center for Real-Time Satellite Mapping at Ohio State University. Tektronix Inc. is a commercial partner for this NASA Spatial Data Display project. Mr. Len Gaydos from USGS, Menlo Park, CA provided the original data for this project. The authors would like to thank Mr. Peter Dotzauer for his software development efforts.

REFERENCES

- Eastman, J.R. 1986. Opponent Process Theory and Syntax for Qualitative Relationships in Quantitative Series. The American Cartographer 13,4: 324-333.
- Horn, B.K.P. 1982. Hill Shading and the Reflectance Map. Geoprocessing 2: 65-146.

CONVERSION OF CONTOURS

B. Shmutter & Y. Doytsher

Technion - Israel Institute of Technology
Haifa , Israel

ABSTRACT

Converting contours is encountered while preparing new maps on the basis of existing maps (for example, conversion of feet into meters). It is assumed that the topography is given by the old contours and the hydrographic network in vector mode, and the contours have assigned elevations.

The new contours are generated by simulating the manual interpolation which skilled cartographers would perform, rather than by forming digital elevation models. This approach seems to be advantageous, since the new contours having been interpolated properly conform to the old ones, which in turn describe the terrain structure satisfactorily.

Generating new contours is performed as follows : Intersecting the hydrographic network with the old contours and assigning elevations to the points of the network elements; Locating for each contour all the neighbouring contours on each side of it; Subdividing the map area into subareas bounded by neighbouring contours of consecutive or equal elevations; Interpolating new contours within the relevant subareas.

INTRODUCTION

There is occasionally a need to insert new contours (generally speaking "isolines" of any nature) between contours of an existing map. Typical examples are converting contours from feet to meters, or changing contour intervals when reducing the scale of the map.

In order to preserve the quality of the given terrain representation it is required that the new contours should conform to the old ones. To meet that requirement, an attempt is made to simulate the manual interpolation, the way in which a skilled cartographer would fit new contours to those existing on the map.

Various aspects of that process are discussed in the following.

PREPARING THE DATA FOR INTERPOLATION

The following assumptions are made as to the data at our disposal.

- The area under consideration is a single map sheet bounded by a quadrilateral (rectangle or trapezoid), which is referred to in the following as the frame of the map.
 - The existing digital terrain description consists of contours given in vector form, strings of points each string carrying an elevation which has been assigned to it. In addition there is a set of vectors defining the hydrographic network. These data are given in terms of planimetry (plane coordinates only).
- The above assumptions are consistent with the procedure of digitizing maps by means of a scanner.

Prior to carrying out the actual contour interpolation several procedures have to be executed to prepare the data.

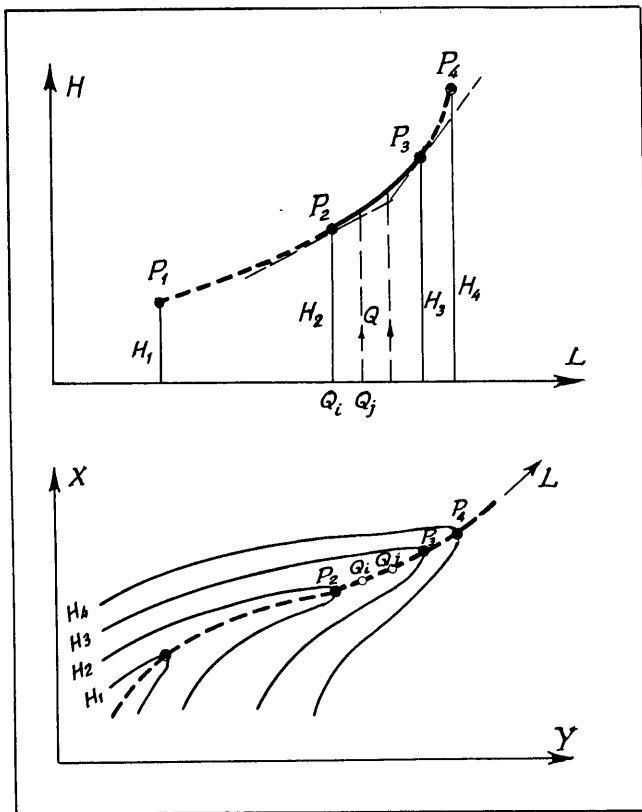


Figure 1. Intersecting a line with contours

Assigning elevations to the points of the hydrographic net

The drainage pattern is an essential component of the terrain description, and as such it has to be incorporated

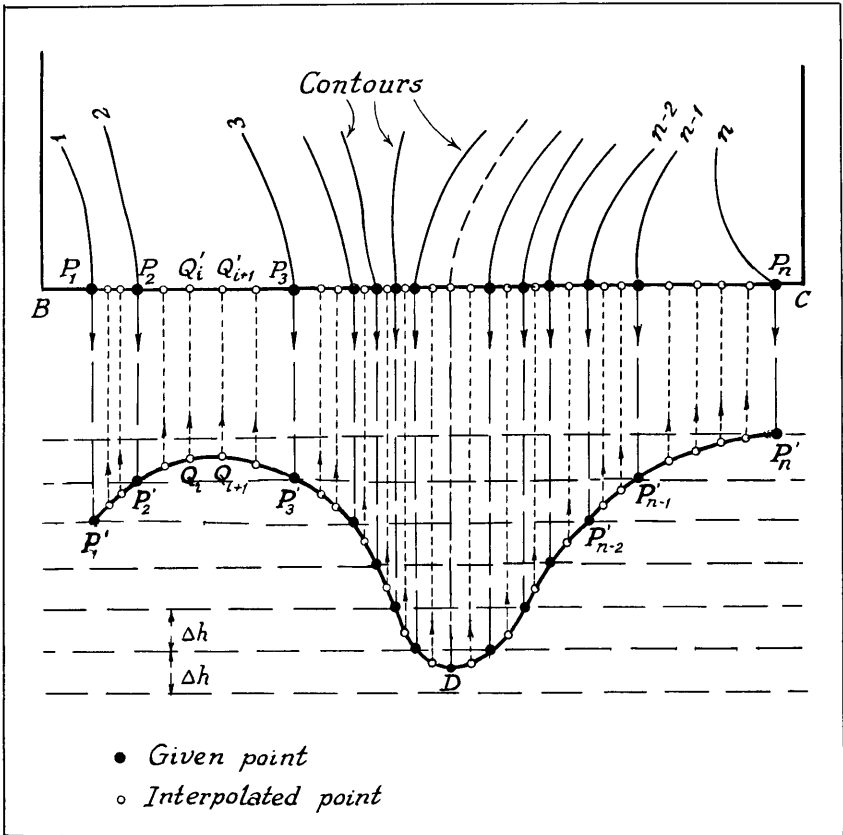


Figure 2. A side of the frame as a terrain profile

in any procedure aiming at the generation of contours. That requires to assign elevations to the points of the drainage lines, which necessitates to intersect these lines with the contours. Figure 1 depicts the intersection procedure.

The line L represents a segment of the drainage pattern. It is defined by points with known plane coordinates. Having selected contours situated in the vicinity of L, intersections are performed between segments of L and the relevant contours, yielding points whose elevations equal those of the respective contours. Usually, four points of intersection are determined (P_1, P_2, P_3, P_4 on figure 1). On some occasions three or two points are located. The points of intersection enable to carry out a non-linear interpolation (when there are more than two such points) to determine elevations for the intermediate points Q_i, Q_{i+1} positioned on the segment P_2-P_3 of the line L. Extremities of drainage lines positioned between contours assume elevations computed from the surrounding contours.

After completing all the intersections and the related

interpolations, the entire hydrographic network becomes determined in terms of x,y,z coordinates.

Intersecting contours with the frame of the map

The frame of the map plays an important role in subdividing the area of the map into strips. To subdivide it properly it is mandatory that contours should terminate at the sides of the frame. Due to errors inherent in the digitized data that condition may not be met. Hence a search is made along the frame of the map to locate extremities of contours which are off the sides. Whenever such an event is encountered the appropriate segment of the related contour is intersected with the respective side, thus coercing the contour to terminate (or start) at the frame.

Sides of the frame constitute terrain profiles. Since the extremities of contours to be generated have to be positioned on the frame as well, the profiles are utilized for determining their locations. For that reason each of the profiles is subjected to a smoothing routine and complemented by additional points which are inserted between the original points of the profile.

Figure 2 illustrates the said above. P_1, P_2, \dots, P_n are the extremities of the existing contours and Q_1, Q_{1+1} are the inserted points. Their locations are chosen in accordance with the intervals between the given points and their elevations are computed by the smoothing routine. If a drainage line intersects a side of the frame, point D on figure 2, the profile is divided into parts, each part being smoothed separately. The upper section of the figure depicts a side of the frame in the x,y plane, the lower represents the smoothed profile.

Determining positions and elevations of peaks

Some of the contours on the map are closed figures. The area bounded by a loop like contour, within which no other contour is present, usually contains a discrete point representing a peak (or the lowest point in case of a depression). In those cases when such points are missing they have to be established from the data of the surrounding contours. Again, such points are needed for proper generation of contours. Various suggestions can be made for how to determine the location and elevation of a peak. The procedure employed here is illustrated by figure 3.

The procedure starts with circumscribing a rectangle of minimal size around the inner contour h_2 . Two profiles directed parallel to the sides of the rectangle can now be formed. Each of them consists of four points: E,F,G,I on one profile and J,K,L,M on the other. Each of those is an extreme point of the two contours with regard to the sides of the rectangle. Parabolas of the third order are fitted to the quadruples of points. The locations and elevations of the extreme points on those curves (P_1, P_2 on figure 3) provide data to position the peak and calculate its elevation.

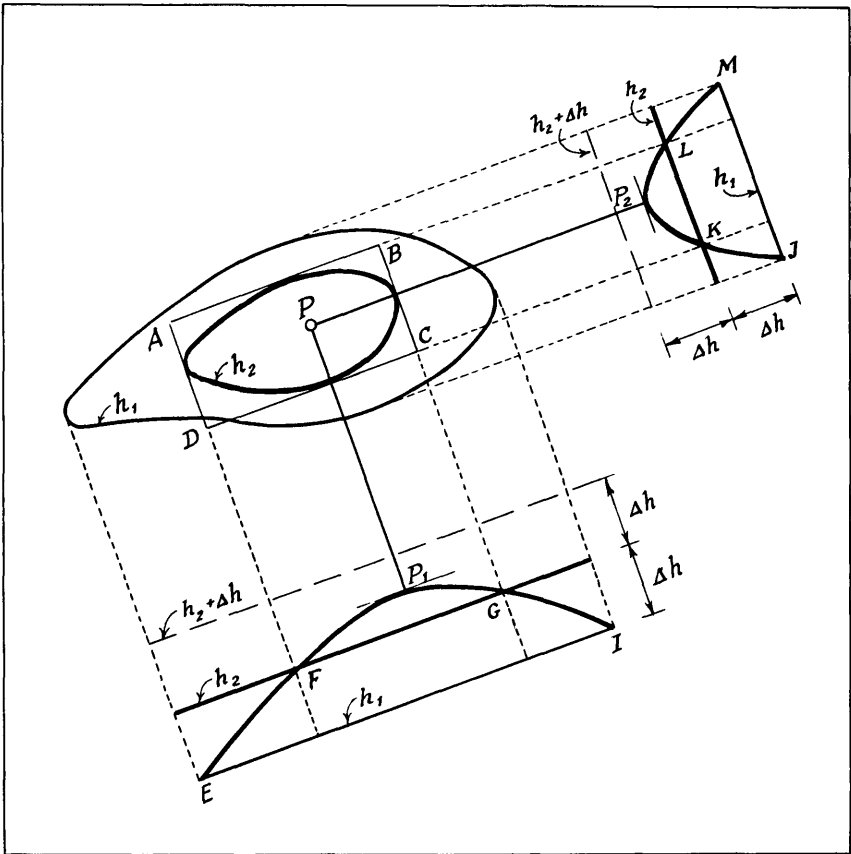


Figure 3. Defining a peak

Subdividing the map area into strips

A vital stage of the process is the subdivision of the map into strips. Contours usually terminate at the frame of the map and eventually close on themselves forming loops. Hence several types of strips have to be considered. Examples of different types of strips are depicted in figure 4.

One strip (no. 1 on figure 4) is defined by the contours a,b and segments of the frame i_{i+1} , k_{k+1} , $D_{D_{k+2}}$. Another is determined by the contours b,c,d and the related segments i_{i+1} , j_{j+1} , k_{k+1} . Regarding loop like contours, the strip is either an innermost loop, loop 3 on that figure, or a strip which assumes a shape of a ring (strip 4) bounded by the contours e and f.

In order to permit defining the strips, proper successions of contour extremities have to be established on each side of the frame. Starting at the side AB of the frame, sequential numbers are assigned to the extremities positioned on that side according to ascending northings. Hence, the

extremities of the contours a,b,c assume the numbers i , $i+1$, $i+2$. The counterpart extremities of those contours assume temporarily the same numbers. Having completed the sorting along the side AB, we proceed with the side BC and assign sequential numbers according to increasing eastings. At that stage the previous numbers are replaced by the new ones. As a consequence of that step it becomes known that the contour c for example starts at the side AB at a point carrying the number $i+2$ and terminates at the point j lying on the side BC. Proceeding in the same manner proper sequential numbers are attached to the contours, so each contour is being identified by the numbers of its extreme points and the sides on which those points are positioned.

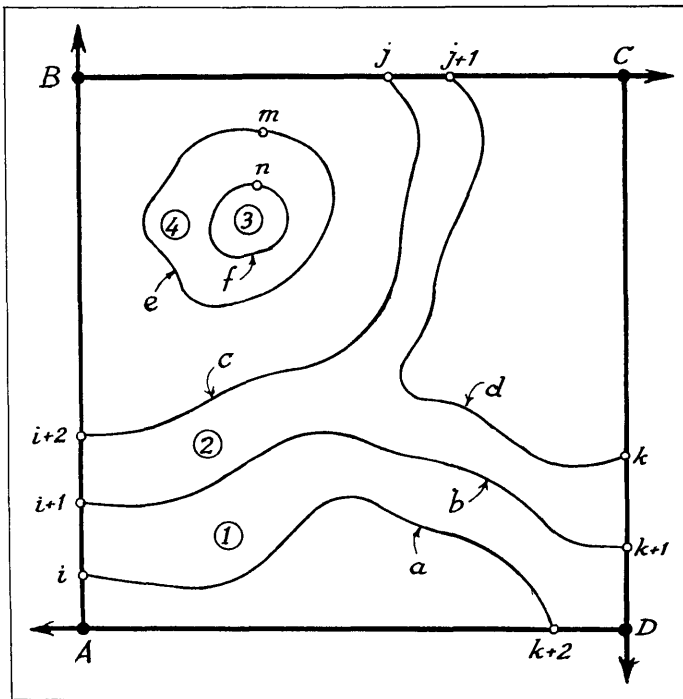


Figure 4. Forming strips

The information about the contours so obtained assists forming the strips. Consider strip no. 1 on figure 4. One of its boundaries is the contour a with extremities i on the side AB and $k+2$ on DA. In order to determine the second boundary, we look for a contour starting at the point $i+1$ which differs in elevation by one contour interval from the previous one, check whether it terminates at a point preceding $k+2$ and carrying the number $k+1$. If that is the case we conclude that the strip 1 is delimited by the contours a,b and the respective segments of the frame. In order to define the next strip (no. 2 on figure 4), it is necessary to find the counterparts of b. Examining the point $i+2$ on

the side AB and the contour associated with it, it is found that the contour terminates at the side BC at the point j . This gives rise to a conclusion that the strip in question is branching. If the next point in the sequence $j+1$ on side BC has the same elevation as j we examine whether the contour starting at $j+1$ terminates at a point carrying the number k . Having been satisfied that that is the case, we establish the fact that the strip in question is bounded by the three contours b, c, d and the respective segments of the frame. Should the point $j+1$ have an elevation not equal to that of j , then we had to examine the point k on the side CD. If the elevation of the contour starting at k were equal to that of the contour b it would constitute a boundary of the strip being formed. In such an event it is apparent that the strip in question contains other contours. The outermost of these are the remaining boundaries of the strip.

Upon completion of examining strips with boundaries terminating at the frame, the loops are being considered. We start with the identification of the inner loops. Such a loop is characterized by the fact that within its area only one point may be present (a peak). Having found an inner loop we inquire if it is encompassed by another loop. If so, a ring-shaped strip is at hand. The said above is exemplified by the contours e, f (see figure 4).

So far a limited number of cases have been discussed. In our opinion these suffice to elucidate the considerations of the strip formation. As a result of the above procedure the entire area of the map can be subdivided into strips. Such a subdivision is shown schematically on figure 5.

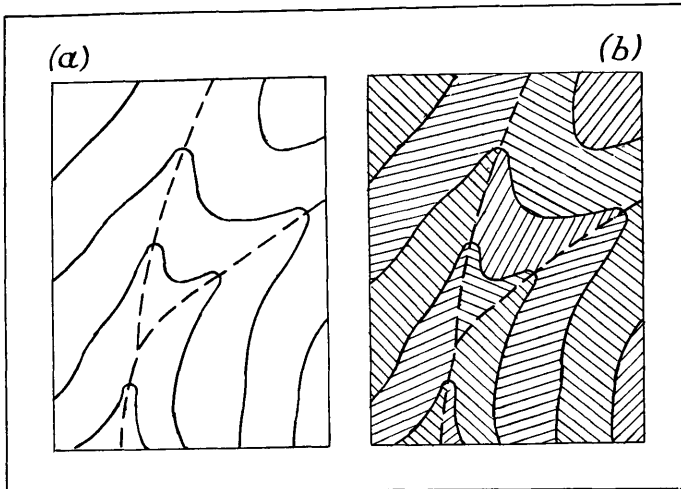


Figure 5. Subdividing the map into strips

INTERPOLATING NEW CONTOURS

A new contour is characterized by its elevation. Examining the sides of the frame it can be found within which strip the contour has to be formed and which are the adjacent contours. To locate the contour, the strip in question is divided into sections in accordance with the shape of the contours delimiting it, while considering the presence of drainage lines crossing it. The shape of the contours are analyzed as follows (figure 6).

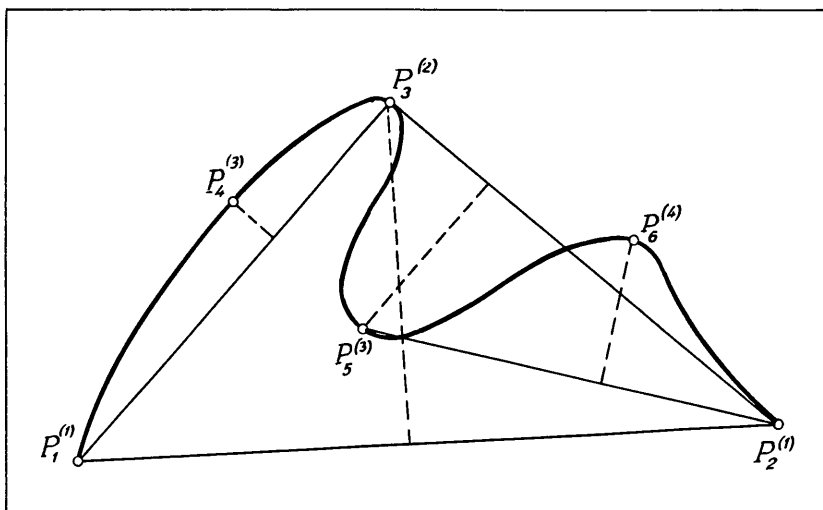


Figure 6. Segmentation of a contour

The aim of the analysis is to identify points which divide the contour into segments having a uniform trend in the xy plane. These are located iteratively by considering the offsets of the points forming the string, in relation to chords of the contour. At the first step, offsets are computed with regard to the chord joining the two extremities P_1, P_2 . The point which yields an offset of a maximal absolute value is assumed as an extremity of a substring, subject to the condition that it exceeds a predefined magnitude. The latter is imposed in order to avoid dividing the contour into too many substrings. Referring to figure 6 a point satisfying the above requirements is found to be P_3 . At the second iteration we examine the offsets related to the chords P_1-P_3, P_3-P_2 . As a result, two additional points P_4, P_5 are found. Lastly, the point P_6 is located with respect to the chord P_3-P_2 . Thus, the contour is divided into substrings: $P_1-P_4-P_3, P_3-P_5-P_4$ and P_6-P_2 . Executing analogous operations with regard to an other contour bounding the strip, establishes pairs of corresponding substrings which enables to divide the strip into sections. The new contour is then interpolated piece by piece in each section separately and joined thereafter to form one continuous line. Eventually, a drainage line passes the strip being processed. In such a case, the suitable segment of that

line defines a boundary of a strip section. Since it is common to two sections, generating the contour in one of them terminates at that boundary and continues from it onward into the other section. That ensures locating the new contours in agreement with the drainage pattern.

The last stage of the process is the actual interpolation. As already said, it is performed within sections of a strip. The main question here is how to select pairs of points, how to associate a point of one string with a point of the other, in order to locate a point on the new contour being generated. Recalling the fact, that the section of the strip is delimited by contours of a nearly uniform trend, it was found convenient to adopt the following approach (see figure below).

Figure 7 depicts a section of a strip bounded by the contours b, c with the adjacent contours a, d on either side. Two curves are fitted to the contours delimiting the section (the lines \bar{b}, \bar{c} on the figure). These two are averaged to form a single curve \bar{m} which represents the general trend of the section. The curve is divided into a number of intervals according to the average number of points of the two contours. A perpendicular to the curve

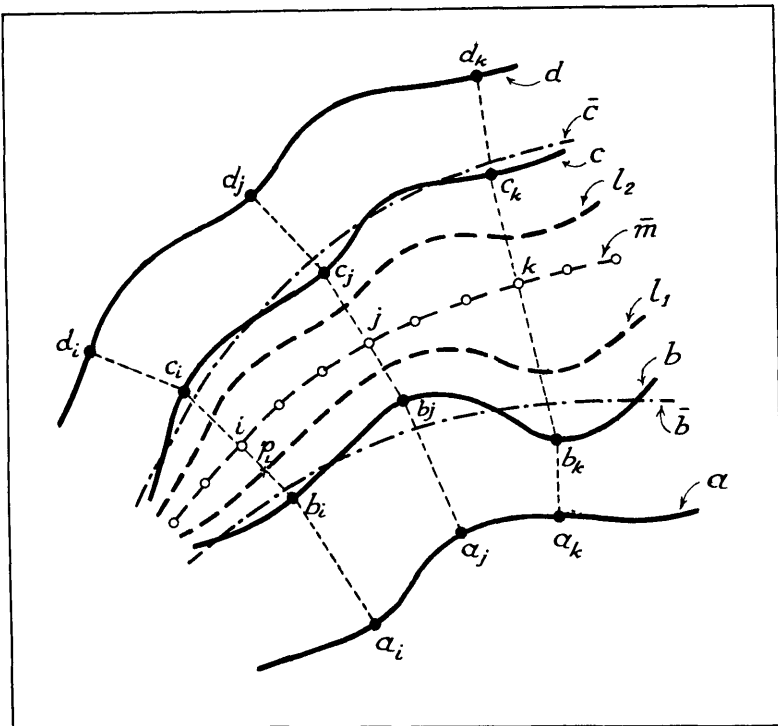


Figure 7. Generating a segment of a new contour

passing through an end point of an interval (the point i on figure 7) intersects the two contours at b_i and c_i . Two more points a_i, d_i are selected from the adjacent strings, the criterion governing their selection being the shortest distances from b_i, c_i to the respective strings. The quadruple of points so obtained forms a X-section of the terrain. From it a point p_i of the sought contour can be computed. To simplify the computation, it is assumed to regard the X-section as if all four points were lying on a straight line. A parabola of the third order is fitted to the X-section and the point p_i of the new contour is located on the line b_i-c_i . The position of p_i is determined from the equation of the parabola, while taking into account its known elevation. A more sophisticated approach would have been to consider the curvature of the X-section (in the xy plane) and to determine the position of p_i accordingly. The feasibility of that approach has not yet been examined.

Applying the above routine to all points on the average curve representing the trend of the strip provides a set of points which constitute a segment of the new contour (l_i on figure 7).

Proceeding in the same manner, segments of the contour are generated in all the sections of the strip yielding the required contour.

CONCLUSIVE REMARKS

Various aspects of the interpolation process have been discussed and essentials of the different stages presented. A comprehensive exposition of the subject would exceed the space allocated for the paper.

It has to be admitted that certain problems related the subject have not yet been solved satisfactorily. These refer to local irregularities (noise) peculiar to contours representing certain types of terrain. Nevertheless, it can be said that on the whole, most of the problems of the computerized simulation of the cartographers skills have been solved adequately.

REFERENCES

- Doytsher Y., Shmutter B., 1986, Intersecting Layers of Information - A Computerizes Solution, Proceedings of AUTO-CARTO LONDON, Vol. 1, pp. 136-145.
- Doytsher Y., Shmutter B., 1987, Digital Urban Mapping, The Cartographic Journal, Vol. 24, pp. 125-130.
- Doytsher Y., 1989, Defining A Minimum Area Rectangle Circumscribing Given Information, The Cartographic Journal (to be published).

RELATIVE ERRORS IDENTIFIED IN USGS GRIDDED DEMS

Dr. James R. Carter
Associate Director, Computing Center, and
Associate Professor, Geography Department
University of Tennessee, Knoxville, TN 37996

ABSTRACT

The 1:250,000 series of gridded DEMs is now complete for the coterminous United States and thousands of 7 1/2 minute gridded DEMs have been released for purchase. As an ever increasing number of persons gain access to these models, it is important that users know of the possible problems, as well as the potential benefits, of using such models. Many of the errors that sneak into DEMs at creation will be removed in the editing stage before the models are released, but some are likely to escape detection. In working with a number of DEMs, the author has made efforts to evaluate how well the models capture the pattern of the land surface. In the larger-scale models, a variety of small, but sometimes significant relative errors have been detected. Relative errors are identified as those instances where one or a few elevations are obviously wrong relative to the neighboring elevations which as a group give an adequate definition of the form of the land surface.

One type of error is associated with the DEMs produced by the Gestalt Photomapper II using NHAP imagery. Working with a DLG-based DEMs produced by digitizing existing topographic maps, the author found three other types of error. In this paper, examples of the various types of errors are shown. Consideration is given to how such errors can be corrected.

INTRODUCTION

The U. S. Geological Survey has released thousands of gridded DEMs for distribution and the 1:250,000 series of DEMs is now complete for the coterminous United States. As an ever increasing number of persons gain access to these digital models, it is important that users know of the possible problems, as well as the potential benefits, of using such models. In the author's work with 7 1/2 minute DEMs and one 1:250,000 DEM, efforts have been made to evaluate how well the models capture the pattern of the land surface. This paper is built on the findings of occasional errors in the DEMs the author has worked with and the many hours of thought the author has given to the question of measuring the accuracy of DEMs.

THE NATURE OF GRIDDED DEMs

There is no standard terminology employed to refer to digital representations of the topographic surface (Carter, 1988), but in the parlance of the U. S. Geological Survey,

the Digital Elevation Model, or DEM, is a gridded array of elevations. Such grids conform to either the graticule of latitude and longitude or to the UTM grid system. Those grids oriented to latitude and longitude are referred to as the arc-second data and are currently produced at either 3 arc-seconds or 1 arc-second. The arc-second grids are non-square reflecting the convergence of the meridians with increasing distance away from the equator. By contrast, the UTM grids are square and are normally referred to as being in the planar format. In the case of the USGS products, the 1:250,000 DEMs are constructed on a 3 arc-second grid and the 7 1/2 minute DEMs are built on a 30 meter square grid (U. S. Geological Survey, 1987).

For many years the only forms of digital elevation data released by the U. S. Geological Survey were the gridded arrays of elevation values as described above. In a new production program, called Mark II, the Survey will be producing representations of the topographic surface as digitized strings of contours in the DLG format (Rinehart and Coleman, 1988, 292). These DLG products will be used to produce gridded DEMs through processes of interpolation and editing.

ERROR AND ACCURACY IN GRIDDED DEMS

With the creation of the DEM product mix, USGS has created a terminology to refer to errors and classifies errors into three types: blunders, systematic errors, and random errors. Blunders are those types of major errors that exceed reasonable limits and can be expected to be removed from DEMs when they are edited prior to release. Based on the source of the DEM and the tested quality of the particular model, a DEM will be classified into one of three accuracy levels. The testing is done by comparing spot elevations in the matrix with a known source and quantifying the fit with the RMSE statistic (Root Mean Square Error). Level-1 DEMs are of the lowest quality. This level of accuracy generally applies to all of those models derived from profiling high-altitude aerial photography, such as was done for many years with the Gestalt Photomapper II instruments. Models that do not meet the lowest level of accuracy are not released for distribution. DEMs designated as meeting Level 2 accuracy standards have been edited to be consistent with existing contour maps and water bodies. DEMs currently being derived from the DLGs will normally be designated as Level 2. Level-3 models are even more accurate and represent a goal to shoot for. DEMs so designated will ". . . have been vertically integrated to insure positional and hypsographic consistency with planimetric data categories such as hydrography and transportation. . . A RMSE of one-third of the contour interval, not to exceed 7 meters in elevation, is the maximum permitted. There are no errors greater than one contour interval in magnitude." (U. S. Geological Survey, 1986, 207).

The author has come to think of the errors in gridded DEMs of two basic types, relative and global, based on the extent of the error. Relative errors are defined as those

instances where one or a few elevations are in obvious error relative to the neighboring elevations which as a group give an adequate definition of the form of the land surface. Global errors are thought of as those situations where the general form of the land surface is adequately defined by the digital data, but the total model departs significantly from the source map or the actual land surface. This treatment of errors is not consistent with the terminology employed by USGS, but it is complementary to their discussions of error and precision.

GLOBAL ERRORS

The focus of this paper is on relative errors, but brief consideration will be given to what the author calls global errors. For users with limited facilities, it is very difficult to identify and assess global errors unless they are very large and obvious. In all of this author's work, little effort has been made to identify global errors for it has generally been assumed that any global errors are insignificant and unimportant for the tasks at hand.

Conceptually, global errors may be thought of as displacements of the entire model along one or more axes. Such displacements may occur relative to the source map if digitized from a map or relative to the actual land surface if derived from field measurements or photos, or the displacements may occur relative to a neighboring map. Any corrections for global errors would involve standard graphics transformations applied to the entire model. These transformations include translation, rotation, and scaling and may need to be applied in a linear or non-linear form.

The only global error this author was able to identify was an error in matching neighboring 7 1/2 minute DEMs. An attempt was made to see how well models would fit together and a FORTRAN program was written to fit models together along their east and west sides. Because the sides of the UTM based DEMs do not consist of a single column of elevations but contain many offsets, the task is not trivial. The author did not continue this line of inquiry and the code was never developed to compare models with their neighbors to the north and south.

The test the author used to evaluate the fit of neighboring gridded DEMs is based on the idea that any distribution of differences between neighboring elevations should be consistent whether the neighbors are in the same DEM or form the boundaries of neighboring DEMs. In the very limited sample examined, some pairs showed no significant difference in the statistics of neighboring columns of elevations between models and within the same model. However, in one case, it appeared that the outer column of elevations of neighboring models was one and the same. In this case one or both models was displaced horizontally. The models so tested were older models of the GPM2 variety and were at the lowest level of accuracy. Presumably, DEMs of Level 2 or 3 accuracy will not display such global errors.

RELATIVE ERRORS

In the larger-scale models, a variety of small, but sometimes significant relative errors have been detected. The author has worked with three different types of DEMs and has encountered errors in each type of DEM. No attempt has been made to develop a typology of errors, but through experience different types of errors have manifested themselves. The author draws upon his experiences with the following DEMs: the W 1/2 Knoxville 1:250,000 in the 3 arc-second format; the Norris, Tennessee, and Thunderhead Mountain, North Carolina and Tennessee, 1:24,000 DEMs derived from NHAP imagery using the Gestalt Photomapper II (GPM2) and released without editing for water bodies; and the Thunderhead Mountain, North Carolina and Tennessee, 1:24,000 DEM based on interpolation from DLG contours.

The author carried out extensive analyses with the 1:250,000 DEM and deemed that the model was essentially error free (Carter, 1987). Subsequently, Houser (1988) examined the model for specific errors and found that there are a few small errors of a very local nature. All of the errors that Houser found show up on a contour plot as a crowding of contours in a small section. Comparing plots of these errors to the original topographic map from which the DEMs were derived, revealed that the errors occurred in sites of very steep terrain where the contours bled together on the original map. In many cases, an index contour label was also found at this site. It is apparent that these small errors are largely the product of too much detail in too small a space and a failure to refine the digital product to account for the finest details. No example of any of the errors in the 1:250,000 model are included in this paper.

GPM2-based DEMs

The author has had the greatest amount of experience with the 1:24,000 DEMs derived from profiling NHAP imagery through the Gestalt Photomapper II. The models the author purchased were some of the earliest released and predated the program of correcting the models to remove blunders in the areas of water bodies before release of the model. The inherent problem of creating proper profiling of DEMs over water bodies was brought home to the author early in his work with the models. The author wrote his own software to process DEMs (Carter, 1983) and because of the limitations of resources worked only with rectangular matrices pulled from the larger DEMs. The first area examined by the author was in the area of Norris Dam, where there is a high flood control dam, a deep valley below the dam, and a flat reservoir above the dam. It was assumed that this complex of topography would be readily identifiable in plots because of the dramatic differences in relief between the land surface and the water. In the first plots made, the dam and valley were readily distinguishable, but the reservoir was not the flat surface it should have been. This led the author to purchase copies of the NHAP imagery used to create the DEMs. Areas of sun glint in the photos on the reservoir above the dam provided the reason why the model was in error, for the GPM2 creates a DEM by mechanically

correlating stereo images and with the sun glint there was no way to objectively correlate the images on the two photos.

The nature of that specific error was not apparent until a larger matrix of elevations was pulled from the DEM and mapped with contours, Fig. 1. This map shows the edges between the patches created in the GPM2 that could not be correlated. To correct for such errors requires a large interactive workstation and appropriate software which most people will not have access to. The Technical Instructions issued by the Geological Survey (1986) describe the many editing and enhancement steps a DEM might be put through before it is to be declared of sufficient quality to be released for distribution. In fairness to the Geological Survey, it should be noted again that the GPM2-based models discussed in this paper were among the first releases of such models and predated many of the editing steps now applied to DEMs. However, this is not to imply that all DEMs being released now will be error free, for as noted in the Technical Instructions, although errors ". . . may be reduced in magnitude by refinements in technique and precision, they never can be completely eliminated." (U. S. Geological Survey, 1986, 2-1).

Fig. 2 displays a similar linear pattern of error but because it occurs along a fairly steep ridge, it is not so apparent. Obviously, the major error in this figure represents the inability to bring patches together in the GPM2. An examination of the NHAP imagery did not provide any clues as to why this error occurred. To the west of this error, another error is found where the contours between 1300 and 1400 meters are compressed in a local area. Again, an examination of the imagery did not reveal the cause of this error. These two errors are relatively minor and might be corrected by a user sketching contours from the published topo quad on a plot of this type and then using an editor to replace individual elevation values with better estimates. The complex error shown in Fig. 3 is even less obvious than those seen previously. This error was considered to be trivial for analyses being conducted by the author. However, in a correlation analysis of synthesized reflectivity values derived from a matrix containing this error with Thematic Mapper data, this error stood out as an extreme departure (Carter, 1989). This revelation pointed out that any error may be significant and all potential errors should be identified.

DLG-based DEM

Having spent many hours with the GPM2-based Thunderhead Mountain DEM, the author was pleased to find that one of the prototypes of the DLG-based DEMs was the same Thunderhead Mountain quadrangle (Berry, Moreland, and Doughty, 1988). The author got a copy of the new Thunderhead Mountain DEM which came from an entirely different source and began to work with it. At the time the model arrived, the author was experimenting with various indices of warp on a square cell formed by the elevations at each corner. These indices were applied to the new DEM and the frequency distributions of

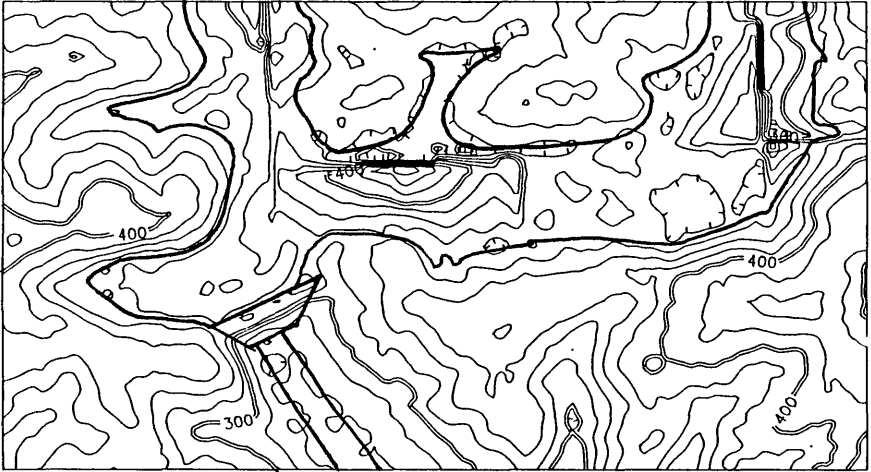


Fig. 1 - Surface II plot using a 20 m contour interval of a 60-Row by 110-Column matrix of elevations featuring Norris Dam and the areas immediately upstream and downstream. The bold lines sketched in by hand show the general form of the dam and reservoir. From the Norris, TN, 1:24,000 GPM2-based DEM before editing. The errors along the edges of some of the GPM2 correlation patches stand out due to their cardinal orientations and artificial nature.

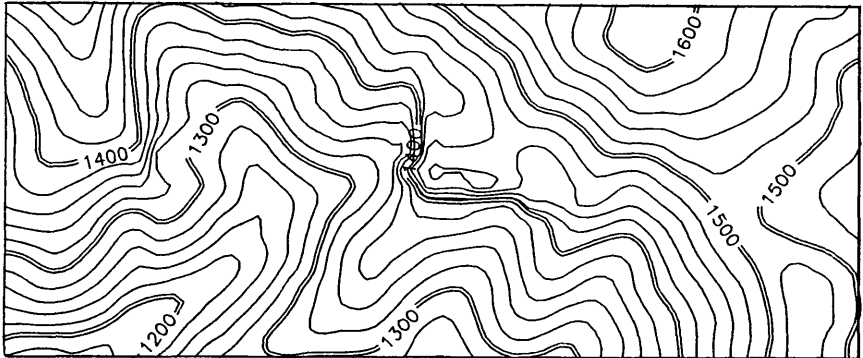


Fig. 2 - Surface II plot using a 20 m contour interval of a 22-Row by 52-Column matrix of elevations from the Thunderhead Mountain, NC and TN, 1:24,000 GPM2-based DEM before editing. The two errors displayed here are less obvious than those in Fig. 1.

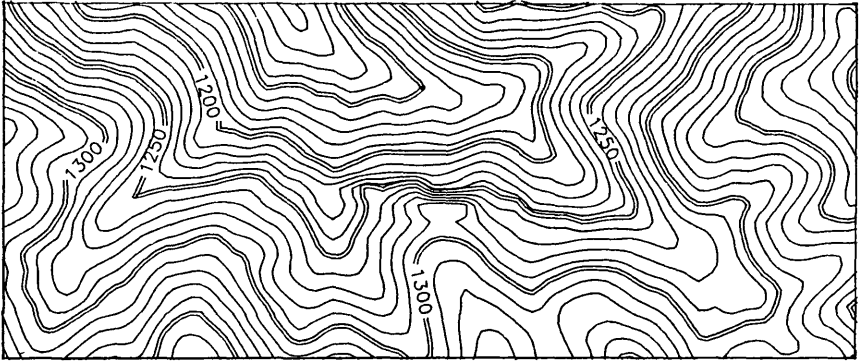


Fig. 3 - Surface II plot using a 20 m contour interval of a 22-Row by 52-Column matrix of elevations from the Thunderhead Mountain, NC and TN 1:24,000 GPM2-based DEM before editing. The complex error shown here is not immediately obvious but proved to be a problem in analyses undertaken by the author.

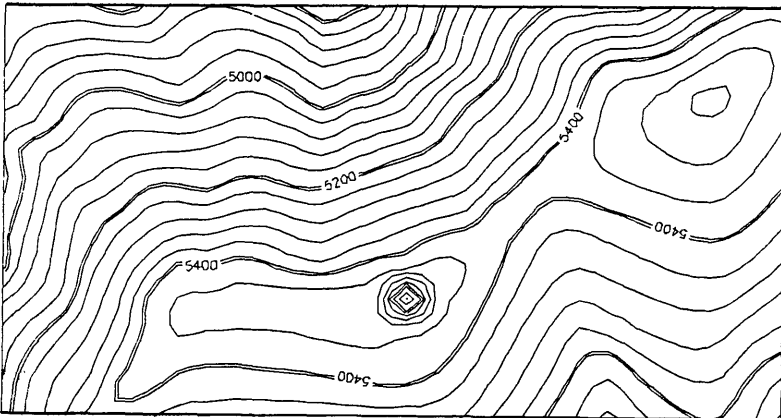


Fig. 4 - Surface II plot using a 40 foot contour interval of a 15-Row by 28-Column matrix of elevations from the Thunderhead Mountain, NC and TN 1:24,000 DLG-based DEM. The spike along the ridge was caused by a single elevation being too high by exactly 200 feet. The peak should be no different from the peak to the northeast.

index values were printed out. Most of the index values conformed to the distributions found in the other DEMs, but a few values were extremely large and well outside the normal distribution (Carter, 1989). The index program was modified to print out the row and column position of all index values above a given threshold. Small rectangular matrices were then pulled at each place where a high index value occurred and contour maps were made of each matrix using Surface II. Each high index value revealed the existence of an error of the type shown in Figures 4 - 6. In total, NINE such errors were found in this DLG-based DEM. In subsequent work with this DEM no other errors or discrepancies have been found. It is assumed that this index departure revealed all of the errors in the DEM, but it would be presumptuous to state that there are no other errors in the DEM.

The most dominant type of error encountered in this DEM was a single peak extending 200 feet above the surrounding lands, Fig. 4. This type of error was found in six places. In all cases, the error occurred where a spot elevation was printed on the topographic map. Listing out the values revealed that in all cases the one elevation in the matrix was exactly 200 feet higher than the spot elevation shown on the map, while all of the neighboring elevations seemed to be correct. The obvious way to correct such errors is to use an editor and replace the value in the matrix, once the error is identified. The small area of the spike probably accounts for the failure to detect the error in the editing procedures.

In two other instances, the errors consisted of a block of elevations being too high by 200 feet. When plotted as contour maps, the erroneous blocks looked like buttes sitting atop a ridge. The errors only became obvious when the contours defining the ridge were compared with the original topographic map. Looking at a listing of the elevations it is fairly apparent that most of the elevations in the block are too high by the same amount. Figure 5 illustrates the erroneous butte that was found along the ridge extending east from Hornet Tree Top in the DLG-based Thunderhead Mountain 1:24,000 DEM.

The 200-foot discrepancies in all of the errors discussed above are obviously a relict to the contour interval on the original topographic quad, for the interval is 40 feet and thus the distance between index contours is 200 feet. One can surmise that such errors come about by tagging index contours, but it is interesting that all of the errors were occurrences where the features were too high by 200 feet. There were no occurrences where the features were too low by 200 feet. It is possible that negative departures occurred in the original model, but were more easy to detect in the editing process and all occurrences were removed.

Another error was detected in the DEM and it was not a simple problem of being off by 200 feet. There is no easy way to describe this error. The map in the left half of Figure 6 shows the contour pattern that Surface II produced from the DEM values. Hand interpolating contours to these

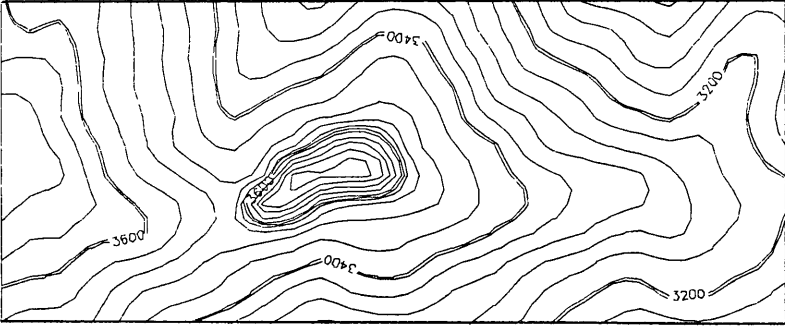


Fig. 5 - Surface II plot using a 40 foot contour interval of a 12-Row by 28-Column matrix of elevations from the Thunderhead Mountain, NC and TN 1:24,000 DLG-based DEM. The butte-like feature at the center of the plot does not exist on the original topographic map although features of this form can be found in nature. The problem appears to be that 7 elevations in the matrix are too high by 200 feet and 2 other values are too high by somewhat less.

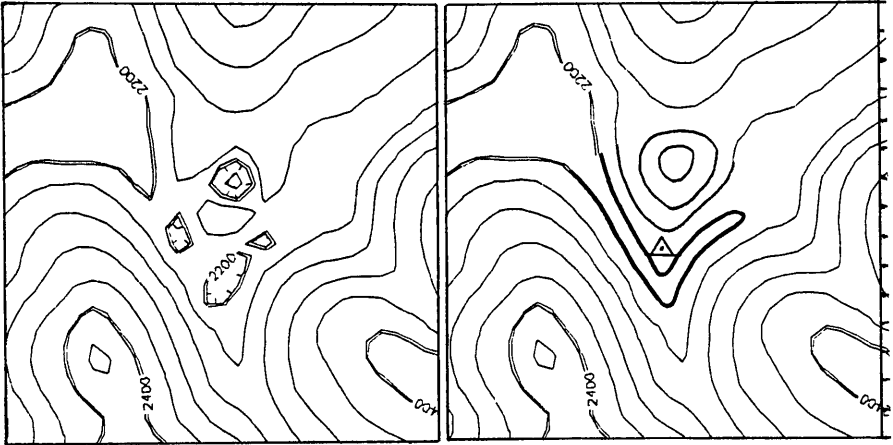


Fig. 6 - On the left a Surface II plot using a 40 foot contour interval of a 16-Row by 16-Column matrix of elevations from the Thunderhead Mountain, NC and TN 1:24,000 DLG-based DEM. On the right is the same plot with the problem area replaced by a sketch of the contour pattern and the BM symbol shown on the original topographic quadrangle map less BM 2215 and the stream name. The ticks on the right represent the spacing of the elevation values.

values produced a similar set of contours, so it is not a problem with Surface II. The map on the right half of Figure 6 shows the same area with the erroneous section cut out and replaced by a sketch of the contour pattern shown on the topographic map. While this is a detailed and complex topographic surface, it's definition is further complicated by having a benchmark and a placename occur in the same area. At the right side of this Figure, ticks have been drawn to show the spacing of the elevation values--30 meters on the ground. It is possible that part of the problem leading to this error is that the detail of the narrow channel is too fine for the spacing of the sample elevations. The author has no suggestions for correcting an error of this type. When the U. S. Geological Survey is able to integrate the hypsographic patterns of the contours with the planimetric detail of the hydrography as will be found in Level 3 models, such errors may become a thing of the past.

CONCLUSIONS

Errors of various types are always going to be with us, in whatever we do. As the digital production activities of the National Mapping program become refined with experience and overt actions, we can expect to see fewer and fewer errors occurring in the datasets entered in to the National Digital Cartographic Database. But it is presumptuous to assume that the Database will ever be error-free. Therefore, it behooves users to become aware of the nature of the types of errors that might exist in any digital database.

In this study, the author shows some of the errors he has encountered in the gridded DEMs he has had an opportunity to work with. Because the author has worked with only a limited sample of gridded DEMs, there may be many types of errors that exist in DEMs created by other processes or DEMs defining landscapes of differing relief and complexity. Colleagues who have worked with DEMs note that they have encountered errors, but no one seems to have shown the nature of the errors they have detected nor has anyone collected representative examples of errors known to occur in gridded DEMs. This paper is offered as perhaps the first of its kind to detail types of errors found in actual DEMs. If readers of this paper have encountered errors of other types, I hope they will make an effort to document those errors so that users can be better informed about the nature of the gridded DEMs.

Acknowledgement: The computing undertaken for this work was carried out on the VAX Cluster at the University of Tennessee Computing Center.

REFERENCES

Berry, Russell D., Deborah K. Moreland, and Eileen F. Doughty, 1988, "Production of Hypsography Digital Line Graphs from the U. S. Geological Survey's 7.5-Minute Map Series," *Technical Papers 1988 ACSM-ASPRS Annual Convention, Vol. 2, Cartography*, 262-71.

Carter, James R., 1983, "Bringing the Digital Elevation Model into the Classroom," *Technical Papers, ACSM 43rd Annual Meeting*, 474-82.

Carter, James R., 1987, "Evaluation of the 1:250,000 Digital Elevation Model for Use in a GIS for the Great Smoky Mountains National Park," *Proceedings, International Geographic Information Systems Conference*, Crystal City, VA, Nov. 1987. (forthcoming)

Carter, James R., 1988, "Digital Representations of Topographic Surfaces," *Photogrammetric Engineering and Remote Sensing*, Vol. LIV, No. 11, Nov. 1988, 1577-80.

Carter, James R., 1989, "Evaluating the Quality and Accuracy of Gridded Digital Elevation Models," *Proceedings, 14th International Cartographic Association Conference*, Budapest, Hungary, August 1989. (forthcoming)

Houser, Jeffrey , 1988, "An Examination of Errors in the W 1/2 Knoxville 1:250,000 DEM," a paper prepared for Advanced Cartography, Geography Department, University of Tennessee, Knoxville.

Rinehart, Robert E. and Earl J. Coleman, 1988, "Digital Elevation Models Produced From Digital Line Graphs," *Technical Papers 1988 ACSM-ASPRS Annual Convention, Vol. 2, Cartography*, 291-299.

U. S. Geological Survey, 1986, "Standards for Digital Elevation Models," U.S. Geological Survey *Open-File Report 86-004*.

U. S. Geological Survey, 1987, "DIGITAL ELEVATION MODELS," National Mapping Program, *Data Users Guide 5*.

THE ARCHITECTURE OF ARC/INFO
Scott Morehouse
Environmental Systems Research Institute
380 New York Street
Redlands, California

ABSTRACT

Arc/Info is a generalized system for processing geographic information. It is based on a relatively simple model of geographic space - the coverage - and contains an extensive set of geoprocessing tools which operate on coverages. Arc/Info is being used in a wide variety of application areas, including natural resource inventory and planning, cadastral database development and mapping, urban planning, and cartography.

The design philosophy and architecture of Arc/Info is described. This includes the spatial data model, the spatial operators, the engineering of the system as a practical software product.

INTRODUCTION

This paper provides a general overview of the philosophy and architecture of the Arc/Info geographic information system. I begin with an overview of the basic approach to GIS system design used by Arc/Info, then review briefly the geographic data model implemented in the system. Any real system is more than just a data model, so the basic geoprocessing tools associated with Arc/Info are introduced. Finally, geographic information systems are complex software systems. I discuss some of the software engineering philosophy and methods that have proved successful in creating Arc/Info.

GENERAL ARCHITECTURE AND APPROACH

There are two basic approaches in GIS development today - the Spatial DBMS approach and the Spatial Tool Kit approach.

In the spatial DBMS approach, the GIS is considered to be a query processor operating on a spatial data base. Users and applications get information by passing a request to the query processor, which navigates the data base to find the answer, which is then returned to the application. In this way, details of the data base implementation are hidden from the application, and other useful functions like concurrency control and crash recovery can be managed. To be useful to the user, however, such a query processor must be very sophisticated - knowing polygon overlay, thematic mapping, attribute modelling, etc. In practice, query processors often simply retrieve geographic data using spatial and attribute keys, leaving more complex geographic modelling and cartographic tasks for the application programmer. Some data modelling problems can be solved within the data base. For example, the polygon overlay problem can be solved by overlaying all data as it is added to the data base. In this way, all queries involving multiple data layers can simply become attribute based queries.

The spatial DBMS approach typically involves an interactive database editor which is used to load and edit the spatial data base. It establishes the necessary topology, spatial indexes, and between layer links necessary for the query processor.

This is the classic Data Base Management System approach to the GIS problem: "How can we extend the (choose one: network, relational, object-oriented) data base approach to support geographic information?" This approach is popular in recent GIS designs (see, for example [Frank, 1982], [Herring, 1987], [Bundock, 1987], and [Carlwood et al, 1987]). It is also pursued by computer scientists seeking to extend relational and object-oriented data base management systems [Stonebraker, 1986].

The principal drawback of the spatial DBMS approach is the difficulty of application development. If the problem cannot be solved by the query processor, then an application program must be written that extracts the relevant information and does the geoprocessing problem itself.

The other basic approach is the application development or tool box approach. The central paradigm of this approach is "application oriented tools operating on objects". It is more closely related to work in document processing and software development environments (e.g. UNIX) and to fourth generation languages than to the DBMS approach. In the tool box approach, we define objects, which are pieces of geography, together with a set of geoprocessing operators for these objects (see figure 1).

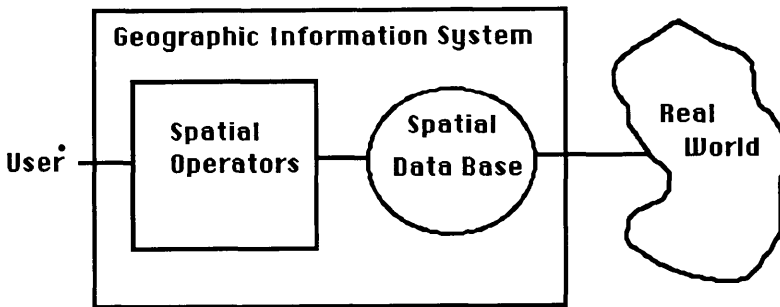


Figure 1: The GIS as a Geoprocessing Tool Box

Objects are stored in a data management system which provides for storage on disk, backup, concurrency control, etc. Operators (tools) are organized into a high level language system which provides a standardized user interface and a mechanism for combining tools into higher level tools.

Unix is one example of the tool box approach. In unix, the

objects are files. A file is simply a string of bytes with a name. Files are organized into directories and stored on disk. Files can contain any sort of data, although some tools may assume files contain text, object code, or executable code. The operators are unix commands. Most unix commands act as file processors, reading in files and writing transformed files. Commands are organized by the command shell. The shell provides the user interface along with a mechanism for writing command procedures (shell scripts). Unix is a powerful text processing and software development system because complex operations can be easily created by combining generic predefined operations.

There are other examples of the tool box approach in mapping applications. Most image processing systems follow this approach with the images as objects and image transformations as being the operators. The Map Analysis Package (MAP) is a geographic information system where the objects are grids and the operators are grid cell analysis commands [Berry, 1987].

These systems illustrate some features that are important attributes of the tool box approach.

Uniform Objects. If operators are to be combined flexibly, they must input and output data in the same format. It should be possible to take the output from any operator and use it as input to any other.

Object Management. There must be a data management system for objects which allows them to be organized and managed with security, backup capabilities, distributed data base functions, etc.

User Interface. There must be an environment which manages the user interface to operators and allows new operators to be easily created from existing ones.

General Operators. Operators should be designed as general purpose functions. This allows them to be combined flexibly for a variety of different applications.

The ARC/INFO geographic information system is based on the tool kit philosophy. It was inspired by earlier developments in unix, the Map Analysis Package [Tomlin, 1983], and the Odyssey geographic information system [Chrisman, 1979][White, 1979].

In ARC/INFO, the objects are vector locational data and the operators are geoprocessing commands for editing, analyzing, and displaying these objects.

THE ARC/INFO DATA MODEL

The ARC/INFO data model, together with its goals, is described in detail elsewhere [Morehouse, 1985]. It will be outlined briefly here. A fundamental goal in the development of the data model is that it perform well in the tool kit approach. This requires a simple yet general data model.

The basic unit of data management in ARC/INFO is the coverage. A coverage defines locational and thematic attributes for map features in a given area. The coverage concept is based on the topological model of geographic information and may contain several types of geographic features. Figure 2 shows the principal feature classes that may be present in a coverage. These feature classes form the basic vocabulary used to define geographic information in a coverage. By varying the types of features contained in a coverage and the thematic attributes associated with features, the coverage can be used to represent many types of map information.

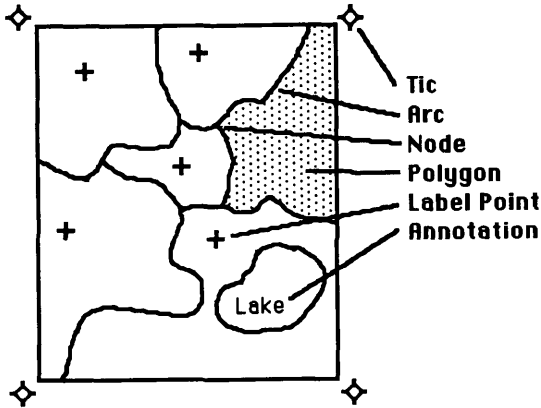


Figure 2: Feature Classes in an ARC/INFO Coverage

Each feature class may have an associated feature attribute table. Each table defines the attributes (called "items") for all features of that class in the coverage. There is a record for each individual feature. The feature attribute tables are an integral part of the coverage and are processed by ARC for all ARC/INFO commands which affect the coverage. For example, when two polygon coverages are overlaid to create a new composite coverage, the polygon attribute tables of the input coverages are joined and written as the polygon attribute table of the output coverage.

ARC/INFO provides a mechanism for the management of very large geographic data bases through the Map Library. The map library organizes data as a set of layers and tiles. Layers are, in most respects, like coverages except that they are partitioned spatially by tiles. The Arc/Info data base is implemented using relational data base modelling techniques. A coverage is defined by a set of relations. Some of the key relations in the coverage are:

```
ARC: (arc#,f-node#,t-node#,l-poly#,r-poly#,xy...xy)
AAT: (arc#,item-1...item-n)
LAB: (label#,poly#,xy)
PAL: (poly#, arc#...arc#)
PAT: (poly#,item-1...item-n)
```

These relations define the geometric, topological, and

attribute values of the various features in the coverage. We have found the relational approach to be very valuable for a number of reasons. First, each Arc/Info tool can choose to access and create coverage relations in the way most appropriate to that tool - there isn't a single method used by all tools to access and update the data base. One example is writing new arcs to a coverage from a bulk data process (e.g. polygon overlay). In the bulk process, some relations, such as the list of arcs around polygons, can be created via more efficient algorithms than would be possible in an interactive editor.

Second, and more importantly, the relational approach allows us to grow the data base schema by simply adding new relations to the coverage model. For example, spatial indexes for all feature classes were added to Arc/Info by simply creating some new relations in the coverage and then teaching the spatial search module how to use them.

THE GEOPROCESSING TOOLS

Given the definition of objects in the Arc/Info data model - coverages, map libraries, tics, arcs, etc. - Arc/Info can be defined as the set of appropriate and useful tools which operate on these objects. This is an open-ended definition. The Arc/Info tool box is intended to grow and develop with GIS technology and with our users needs.

The Arc/Info tools operate at a variety of levels. There are tools which operate on entire map libraries, others which operate on coverages, and finally tools for manipulating individual features.

Map library tools. These tools all operate on map libraries (see figure 3) and are collected as the librarian function.

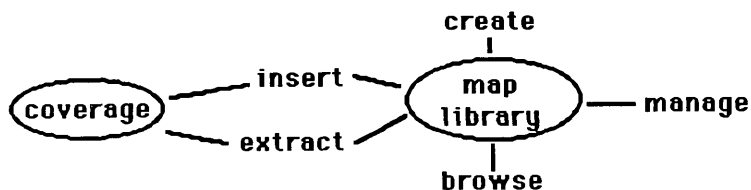


Figure 3: Map Library Tools

The librarian defines and manages map libraries. The librarian has a number of tools which operate on libraries. Geographic data in the map library is managed like software in a source code management system. To perform an update, the relevant layers and area of modification are extracted to an operators workspace where the geographic data is edited and the edits verified. The verified data is then reinserted into the map library.

The librarian manages this entire process as a transaction on the map library and prevents simultaneous extraction of the same layer and affected area for modification by other users of the library.

This extract/insert approach to transactions on geographic data bases is necessary because revisions to geographic data generally involve long highly interactive processing and verification of the data. In most ways, update transactions on a geographic data base are more like transactions on a source code library than transactions on a tabular data base [Aronson, 1989].

The librarian also provides browse functionality. In this situation, most query and display tools which operate on coverages in a read-only fashion can also operate on entire map library layers as if they were a single seamless coverage.

Coverage Tools. These tools all operate on entire coverages, managing all feature classes and associated attributes in the coverage as a single unit (see figure 4).

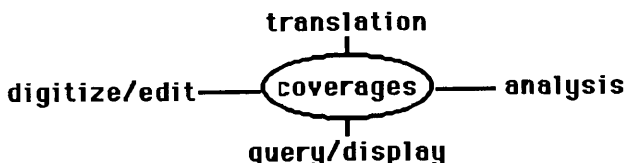


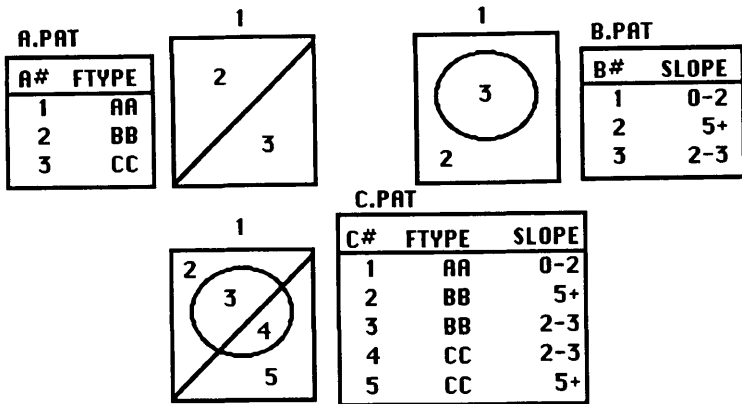
Figure 4: Coverage Tools

These tools can be loosely grouped into four categories: translation, digitize/edit, analysis, and query/display.

The translation tools perform the conversion of data between a variety of spatial data formats and Arc/Info coverages. Translators which are presently supported include DLG-3, DXF, IGES, Moss, SIF, ascii, and various raster formats.

The digitize/edit tools support creating and editing coverages. The primary tool (or collection of tools) here is Arcedit. Arcedit is an interactive graphics editor for coverages. Other tools exist to support bulk generation of topology, data verification, form driven attribute data entry, and a number of other tasks necessary in creating a geographic data base.

The analysis tools perform spatial analysis functions involving one or more coverages. Generally the results of the analysis are written as a new coverage or as additional attributes on an existing coverage. The classic example of this type of tool is the polygon overlay or "spatial join" tool. This class of tool takes two coverages, finds all intersections between features and writes the resulting integrated coverage as a new coverage (see figure 5).



UNION A B C

Figure 5: Coverage Overlay

Arc/Info has an extensive set of spatial operators at the coverage level. These include:

coverage overlay:

- polygon on polygon
- point on polygon
- line on polygon

thiessen polygon generation

contour interpolation

buffer zone generation

network allocation

map projection and coordinate transformation

rubber sheeting

generalization

feature selection and aggregation

arithmetic and logical attribute combination

The Arc/Info data model has been specifically designed to support these coverage level spatial analysis tools as well as query and edit tools which operate at the individual feature level.

The query/display tools are used to view the geographic data base and to perform ad hoc queries on the data base. Tools are provided to define and edit cartographic symbols, to scale and position map graphics, to associate cartographic symbols with geographic feature attributes, and so on. As a brief example of these tools, imagine the problem of selecting and drawing all roads in a given area which pass through hardwood forests. The Arc/Info tools which could be applied to this query are:

```

reselect forest polygons type = 'hardwood'
reselect road lines overlap forest polygons
arclines roads type type.symbol

```

The first operator performs an attribute selection on polygon features in the forest coverage. The second operator is a

overlap query which finds all line features in the road coverage which overlap the previously selected forest polygon features. Finally, the third operator displays the selected roads using cartographic symbols derived from the road type attribute.

Nearly all query/display tools operate on map library layers as well as individual coverages.

Feature level tools. These tools operate on individual features within a coverage (see figure 6).

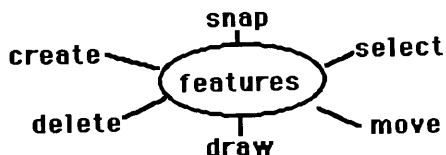


Figure 6: Feature Tools

The primary collection of these tools in Arcedit, the Arc/Info coverage editor. Arcedit provides tools for selecting features, then modifying them in various ways.

THE USER INTERFACE

The Arc/Info user interface can be defined at two levels - the base user interface and the application-oriented user interface. The application-oriented user interface is built on top of the base interface using the Arc Macro Language (AML). Figure 7 illustrates this concept.

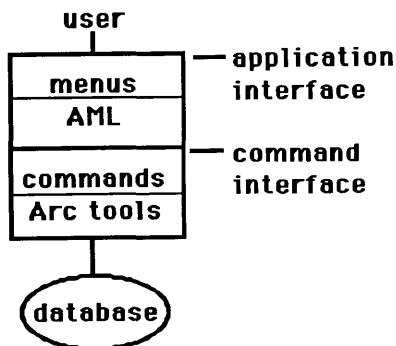


Figure 7: The Arc/Info user interface

The base user interface is a simple command language similar in purpose to operating system command languages found in unix, MS-DOS, VMS, and other operating systems. It is based on the paradigm of tools and objects. Each tool in Arc/Info has an associated command in the Arc command language. This language is very simple, consisting of a verb followed by one or more objects or command qualifiers. For example, the command to invoke the polygon overlay tool is:

intersect <in_cover> <intersect_cover> <out_cover>

(where <in_cover>, <intersect_cover> are names of two existing coverages and <out_cover> is the name of a new coverage to create with the results.)

This level of the interface is designed for generality, extensibility, and flexibility. Commands can be either entered interactively from a keyboard by the user or can be supplied from AML procedure or menu. The command language can also easily grow through the addition of new commands and command qualifiers.

The application-oriented user interface is built on top of the base command interface using the AML language. AML is a procedural language interpreter designed specifically for Arc/Info. It has all of the features typically associated with operating system command languages, such as named variables, flow of control directives, numeric and string operators, and so on. AML also defines a set of user interface objects. These include pull down menus, pop up menus, and forms. These objects can be used to develop a user interface which is designed for a specific application or to provide a non command driven user interface.

THE SOFTWARE ENGINEERING APPROACH

Any GIS is a significant software engineering problem. To be useful, any GIS has to be well engineered. A clever data model or powerful user interface is useless unless the software performs correctly and will work effectively with large collections of geographic information. Software engineering issues are central to the viability of any GIS; they are also very interesting problems in their own right. Geographic Information Systems are ideal software engineering test cases - they involve database, graphics, computational geometry, user interface, and operating system technology. All of these technologies as well as geographic data modelling and analysis and user requirements are experiencing rapid and continual change.

To thrive in this environment, Arc/Info has been designed as a system which can grow and change. It is not a static system which meets a fixed set of preordained specifications. Arc/Info 5.0 is very different from Arc/Info 4.0 of just two years ago, Arc/Info 6.0 will be different again. The central goal in engineering Arc/Info has been to develop an architecture and programming methodology which supports this process. Once you realize that a system must evolve over time a number of other principles follow [Meyer, 1988].

The system must be maintainable. It will be continually modified, extended, and optimized.

The system must be portable. Who knows what the computing/operating environment of the future will be?

The system must be as simple as possible. Simple systems can evolve much faster than complex systems.

The system must be reusable. Code and algorithms must be designed in a way which supports reuse in future, unforeseen applications.

Clearly, the system must also be expandable and correct. Designing for correctness in an evolving system is different than ensuring that a system functions correctly for a single fixed design.

To accomplish these goals, we have adapted the techniques of modular software design and development [Parnas, 1972] in Arc/Info.

We organize all software development around the concept of the module. A module is simply a collection of routines which work together to define a data structure or to perform some function. Modules are entirely self-contained - the code within one module only interacts with code outside the module by well defined function calls. We have identified a number of module types in Arc/Info:

data structure module - defines and implements the behavior of a data structure (e.g. BITSYS - a bitmap manager).

device interface module - define and implement the behavior of a virtual hardware device (e.g. DIGSYS - the digitizer interface).

processing module - define and implement a functional process. This can either be a generic process (e.g. SRTFIL - a disk based sorter) or a specialized process (e.g. OVRSEG - find all intersections between two, potentially huge, sets of line segments).

program module - define and implement an executable program. Typically defines the user interface and functionality of a high level Arc/Info function (e.g. ARCPLOT - the cartographic display and query system).

Each module is typically the work of a single programmer and is designed as an independent unit. Modules only depend on the functional behavior of other modules that they may use. This means that the internal workings of a module can be freely replaced or extended. For example, we have replaced the internal workings of the segment intersection module a number of times without affecting the modules which use it (other than increased performance and reliability).

This modular software engineering approach, together with the simplicity and extendibility of the basic Arc/Info data model are what allows us to continually grow Arc/Info as a software product.

CONCLUSION

Two popular approaches to GIS design are the spatial database management system and the geographic tool box. Arc/Info is an example of the tool box approach. The data model of Arc/Info is based on a combination of the topological network approach for locational information with the relational approach for feature attributes. Arc/Info has an extensive

set of tools which can operate on this data model. Users can interface with the system either at the basic tool level or through applications and interfaces layered on top of these tools. The primary goals in the development of Arc/Info as a software system has been generality and extendibility. All aspects of the system from the data model to the user interface to the internal engineering of the system have had these goals in mind.

This approach and GIS architecture has been very successful. Arc/Info is presently in production use at over 1000 sites worldwide. It is being used for a wide variety of applications including natural resource planning, cartography, tax assessment, and facilities management. The system is a mature system which will continue to evolve and grow to support the changing needs of our users and the GIS community as a whole.

REFERENCES

- Aronson,P. (1989), "The Geographic Database - Logically Continuous and Physically Discrete", Proceedings, Auto-Carto 9, Baltimore, Md.
- Berry,J. (1987), "Fundamental Operations in Computer-Assisted Map Analysis", International Journal of Geographical Information Systems, v.1, n.2, p. 119-136.
- Bundock,M. (1987), "An Integrated DBMS Approach to Geographic Information Systems", Proceedings, Auto-Carto 8, Washington, D.C., p. 292-301.
- Carlwood,G., G. Moon, and J. Tulip (1987), "Developing a DBMS for Geographic Information: A Review", Proceedings, Auto-Carto 8, Washington, D.C., p. 302-315.
- Chrisman,N. (1979), "A Many Dimensional Projection of Odyssey", Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.
- Frank,A. (1982), "MAPQUERY: Data Query Language for Retrieval of Geometric Data and their Graphical Representation", Computer Graphics, 16, p.199-207.
- Herring,J. (1987), "TIGRIS: Topologically Integrated Geographic Information System", Proceedings, Auto-Carto 8, Washington, D.C., p. 282-291.
- Meyer,B. (1988), Object-oriented Software Construction, Prentice-Hall, London.
- Morehouse,S. (1985), "ARC/INFO: A Geo-Relational Model for Spatial Information", Proceedings, Auto-Carto 7, Washington, D.C., p. 388-397.
- Parnas,D. (1972), "On the Criteria to Be Used in Decomposing Systems into Modules", Communications of the ACM, vol. 5, no. 2, pp. 1053-1058.
- Stonebraker,M. (1986), "The Design of POSTGRES", Proc. 1986

ACM-SIGMOD Conference on Management of Data, Washington, D.C.

Tomlin, C.D. (1983), "Digital Cartographic Modelling Techniques in Environmental Management", Doctoral Dissertation, Yale University, School of Forestry and Environmental Studies, New Haven, Connecticut.

White, D. (1979), "Odyssey Design Structure", Harvard Library of Computer Graphics, 1979 Mapping Collection, Vol. 2, pp. 207-215.

THE COMBINATORIAL COMPLEXITY OF POLYGON OVERLAY

Alan Saalfeld
Bureau of the Census

ABSTRACT

The number of elementary connected regions arising from polygon overlay of two or more map layers is an important value to have in planning for data storage and in making processing time estimates for overlay applications. That number may be computed directly from the line graphs of the two (or more) layers and from the intersection graph(s) of those line graphs. A formula for that computation is derived using tools of algebraic and combinatorial topology which relate the connectivity of a union of sets to the connectivity of the sets themselves and their intersection. The result and the formula may be stated as follows:

Suppose X is the line graph (1-skeleton) of a map. Regard X as embedded in the plane. Let $r(X)$ be the number of regions of the plane separated by X . Then $r(X)$ is the number of connected components in the planar complement of X ; $r(X)$ is also one more than the maximum number of independent cycles in the graph X ; and $r(X)$ is easily computed using standard graph traversal techniques for counting independent cycles. Let $c(X)$ be the number of connected components of X .

If A and B are the line graphs of maps to be overlaid, then $A \cup B$ is the line graph of the overlay; and:

$$r(A \cup B) = r(A) - c(A) + r(B) - c(B) - r(A \cap B) + c(A \cap B) + c(A \cup B)$$

All of the values on the right hand side of the equation can be readily computed using standard graph traversal and line intersection algorithms to obtain the desired value, $r(A \cup B)$, the number of regions after overlaying.

1. INTRODUCTION

The fundamental naive combinatorial question regarding polygon overlay is the following: If I overlay a map of n regions on another map of m regions, how many regions are there in the composite map? The possible answers are: any number that is not smaller than $\max\{m, n\}$. Hence, the answer that we give cannot be a number or even a bound. We relate the number, instead, by an exact formula, to the number and kind of line intersections that occur. In so doing, we transform the problem into one that is more amenable to analysis and to establishing constraints. In this paper we present some methods and results of algebraic topology that illustrate the nature and the methods of dimensional duality for addressing some of the global questions in mathematical cartography. We do not pretend to develop theory of algebraic topology in any detail here—indeed, to arrive at our small result, we must skim over a great deal of mathematics. The interested reader is directed to Henle [1] for more of the topological and combinatorial details and to Hu [2] for a more complete exposition of algebraic concepts.

This paper introduces and describes a limited number of tools of algebraic topology—a sufficient number to derive the formula that relates intersections to the number of regions of the overlay.

2. PRELIMINARIES: CHANGING TOPOLOGY TO ALGEBRA

2.1. Basic Concepts in Algebraic Topology

Algebraic topology is the area of mathematics that examines algebraic properties of algebraic objects derived from topological spaces. Spaces which are topologically equivalent have the same collection of algebraic objects associated with them; and mappings between topological spaces have associated with them mappings between the corresponding algebraic objects. Topological problems are converted to algebraic problems under the described association (formally this association is called the functor from the category of topological spaces and continuous functions to the category of groups and group homomorphisms or the category of rings and ring homomorphisms or some other algebraic category).

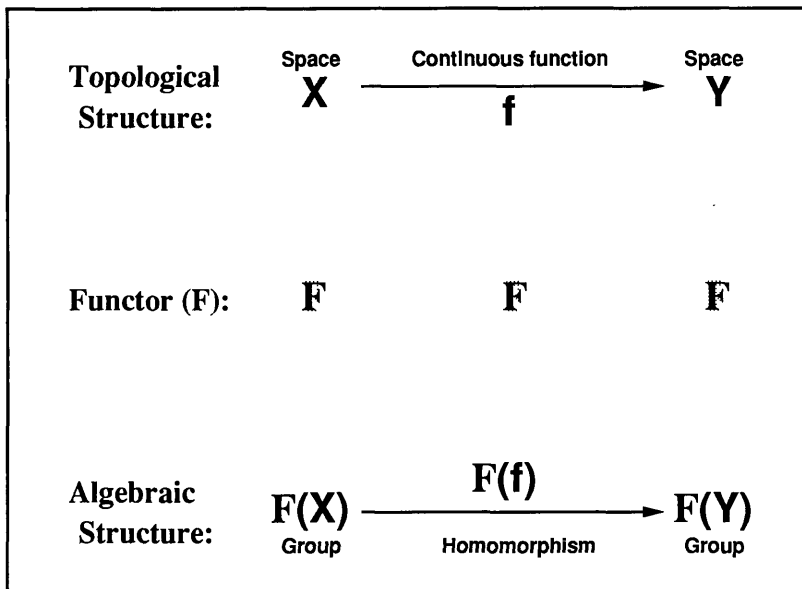


Figure 1: A functor converts topological structure to algebraic structure

Inevitably the algebraic invariants of topological spaces and topological functions cannot retain all of the topological information of the spaces and functions themselves. Often, for example, the algebraic objects are finite, or finitely generated and enumerable, while the interesting topological objects are uncountably infinite. Nonetheless, the reduction of information content to finite or finitely generated sets is precisely the transformation we need to operate with our mathematical model of a-map-as-a-continuum on a computer, which is a finite machine. The map, which has infinitely many points, is partitioned into finitely many cells, which we call 0-cells, 1-cells, and 2-cells depending on their dimension. Those finitely many cells are used to build algebraic structures called chain groups, one group for each relevant dimension; and algebraic boundary operators (homomorphisms) are defined between those groups which capture the essential topological boundary relations among the 0-cells, 1-cells, and 2-cells. Each element of the n -dimensional chain group is a formal linear combination of independent symbols, one symbol for each different n -cell.

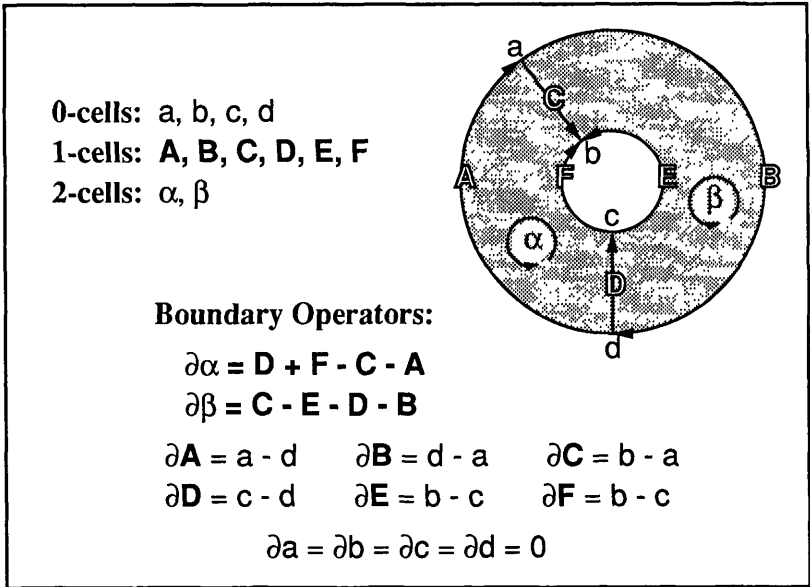


Figure 2: Cell decomposition of annular region, associated group generators, boundary operators, and typical elements

The chain groups and boundary homomorphisms depend on the choice of cell decomposition of the space; and a map may usually be decomposed into cells in various ways.

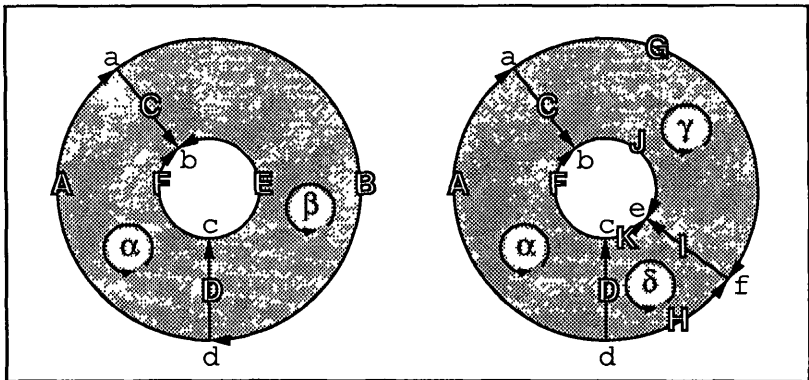


Figure 3: Two different cell decompositions of a region

2.3. Building Composite Algebraic Structures From Elementary Algebraic Structures on Topological Spaces

New groups, called homology groups, may, in turn, be derived from the chain groups by forming quotient groups of distinguished subgroups of cycles and boundaries of the chain

groups. These homology groups surprisingly do not depend on the cell decomposition of the topological space, but on the space itself! That is, two different cell decompositions of the same space will produce two different chain groups, but the distinguished subgroups of the two chain groups will always, in turn, produce the same (up to isomorphism) collection of homology groups.

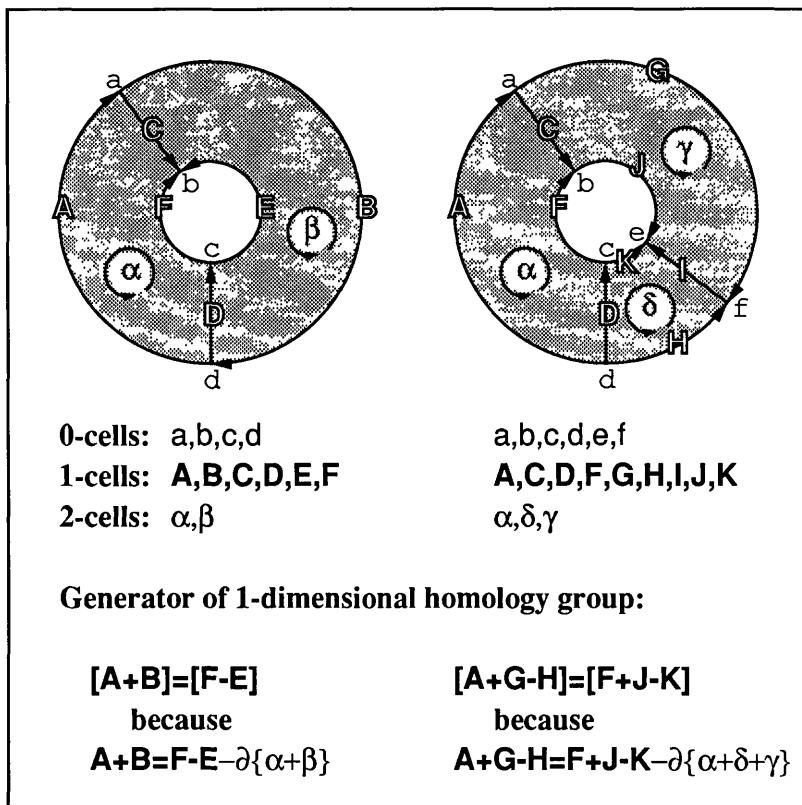


Figure 4: Different cell decompositions yield same homology

Now let's look at the underlying significance of homology groups, and we will describe without proof the structure of homology for many topological spaces, including plane graphs (i.e. the linework of our cartographic objects).

2.4. Some Examples of Homology Groups

Homology groups describe the connectivity structure of the topological space. For maps represented by a full complement of 0-cells, 1-cells and 2-cells, the homology groups are uninterestingly trivial because the full cell structure adds up to a space which is topologically trivial—i.e. equivalent to a rectangle or (if it is a world map) equivalent to a sphere. All homology groups of a rectangle are 0 except the 0-dimensional group, which is \mathbb{Z} , a single copy of the integers. We write $H_0(R) = \mathbb{Z}$.

For the sphere, we have $H_0(S) = H_2(S) = \mathbb{Z}$, and for all i different from 0 and 2, $H_i(S) = 0$.

The somewhat more useful homology groups are those of the line network (sometimes called the 1-skeleton) of the map. The 1-dimensional homology group measures simple connectivity (or lack thereof) of the topological space; and the graph network has many cycles and thus is not simply connected (“Simply connected” means that any loop can be shrunk continuously to a point without leaving the space.) The plane and the sphere are both simply connected. The annular region of figures 2 to 4 is not simply connected, hence H_1 of that region is not 0.

The following are useful summaries of how homology groups behave for the line graph network of a map and what they show about that network:

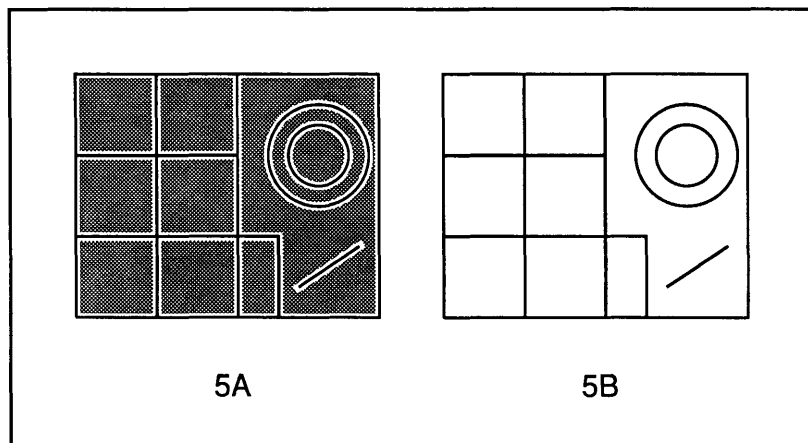


Figure 5: A map (A) and its line graph network (B).

For a topological space consisting of the linework of a planar graph (such as shown in figure 5B), the homology groups have the following structure:

$H_0(X) = \mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z} \oplus \dots \oplus \mathbb{Z} \oplus \mathbb{Z}$, n copies of \mathbb{Z} , the integers, where n is the number of connected components of X . In the case shown in figure 5B, $n = 4$.

$H_1(X) = \mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z} \oplus \dots \oplus \mathbb{Z} \oplus \mathbb{Z}$, m copies of \mathbb{Z} , the integers, where m is the maximum number of independent cycles of the graph X (“cycles” in the graph-theoretic sense, “independent” in the algebraic sense—no non-trivial linear combinations of these elements are zero.) In the case shown in figure 5B, $m = 10$, and a collection of generators for those cycles (in the graph sense) would be sums of the appropriately signed edges making up the outer boundaries of the ten regions shown in figure 5A. Notice that there are far more than 10 different cycles on the graph. What the homology group captures with its algebraic structure is the dependence relations of all of those infinitely many cycles. The homology group is more than just a count of how many independent cycles there are!

$H_i(X) = 0$ for all $i > 1$ since the line graph X has no 2-dimensional or higher dimensional elements which might generate cycles in the homology sense.

Loosely speaking, then, H_0 counts connected components of the line network, and H_1 , though it is the homology group of the line network itself, also counts fundamental (interior) regions delimited by the line network.

3. ALGEBRAIC TOOLS

3.1. Semi-exact Sequences and Exact Sequences

Algebraists have developed a standardized shorthand notation to describe essential structure of interesting subgroups and quotient groups: They have converted objects into homomorphisms and into sequences of homomorphisms in order to treat objects and homomorphisms with the same tools and operators. The tools focus on two important subgroups of a homomorphism, the kernel (\ker) and the image (Im).

If $\Phi : G \rightarrow K$ is a homomorphism of groups then:

$$\ker(\Phi) = \{g \in G \mid \Phi(g) = e_K, \text{ the identity of } K\}$$

and

$$\text{Im}(\Phi) = \{k \in K \mid k = \Phi(g) \text{ for some } g \in G\}.$$

If a sequence of two or more homomorphisms may be composed with each other because the appropriate domains and ranges match, then we may examine the relation of the image of a homomorphism to the kernel of its successor:

$$\xrightarrow{\Phi_{i+1}} G_i \xrightarrow{\Phi_i} G_{i-1} \xrightarrow{\Phi_{i-1}} G_{i-2} \xrightarrow{\Phi_{i-2}} G_{i-3} \xrightarrow{\Phi_{i-3}}$$

If the image $\text{Im}(\Phi_{i-k})$ is contained in the kernel $\ker(\Phi_{i-k-1})$ for all meaningful values of k , then we say that the above sequence is semi-exact.

If the image $\text{Im}(\Phi_{i-k})$ is equal to the kernel $\ker(\Phi_{i-k-1})$ for all meaningful values of k , then the sequence is exact.

The two fundamental results on sequences of chain groups and induced groups, given without proof, are the following:

1. The boundary operators for chain groups always yield semi-exact sequences.

Elements that lie in the kernel of a boundary operator have zero boundary; and we call them cycles. Elements that lie in the image of the boundary operator are called boundaries (because they are boundaries of something!) Cycles that are not boundaries generate the homology groups, which describe the extent to which the semi-exact sequences induced by the boundary operators fail to be exact.

2. Homology groups may be embedded in natural exact sequences whose homomorphisms are induced by the boundary operators and inclusion maps

One such exact homology sequence is the Mayer-Vietoris Exact Homology Sequence described in the next section.

3.2. The Mayer-Vietoris Exact Homology Sequence

The Mayer-Vietoris Exact Homology Sequence relates the homology groups of the union and intersection of two “nice” topological spaces to the homology groups of the spaces themselves by embedding all the groups in an exact sequence:

$$\cdots \rightarrow H_i(A \cap B) \rightarrow H_i(A) \oplus H_i(B) \rightarrow H_i(A \cup B) \xrightarrow{\partial} H_{i-1}(A \cap B) \rightarrow \cdots$$

Knowing that a sequence is exact, and knowing some of its groups, one may often deduce the missing groups. That is the approach that this exposition will utilize. We will not

worry about the way in which the exact sequence is defined. The interested reader is referred to Hu [2] for a full explanation of the Mayer-Vietoris Sequence and sufficient conditions on the topological spaces A and B to guarantee exactness of the sequence.

4. USEFUL PROPERTIES OF HOMOMORPHISMS AND EXACTNESS

4.1. Rank of a commutative group

All of our homology groups are commutative and are finitely generated. Suppose that we have any commutative group that is finitely generated. Then the theory of groups tells us that the commutative group may be regarded as a direct sum of a number n of copies of the integers Z , $Z \oplus Z \oplus Z \oplus \dots \oplus Z \oplus Z$, plus T , the torsion or finite subgroup of the larger group consisting of all elements of finite period.

The value n totally and uniquely determines the algebraic structure of the torsion-free part of this direct sum. The number n is called the rank of the group: and for any group homomorphism Φ , the rank has the following nice additive property:

$$\Phi : G \longrightarrow K$$

$$\text{rank}(G) = \text{rank}(\ker(\Phi)) + \text{rank}(\text{Im}(\Phi))$$

We will use this property to prove an important lemma.

4.2. Telescoping Lemma

The next lemma is the key to constructing a missing group in an exact sequence of groups:

Lemma: Suppose that the sequence given below is exact and that each group G_i has rank n_i .

$$0 \xrightarrow{\Phi_{i+n}} G_n \xrightarrow{\Phi_n} G_{n-1} \xrightarrow{\Phi_{n-1}} \dots \xrightarrow{\Phi_3} G_2 \xrightarrow{\Phi_2} G_1 \xrightarrow{\Phi_1} 0$$

where Φ_{n+1} and Φ_1 are the zero homomorphisms. Then consider the following alternating sum:

$$\begin{aligned} (-1)^n \text{rank}(G_n) + (-1)^{n-1} \text{rank}(G_{n-1}) + \dots + \text{rank}(G_2) - \text{rank}(G_1) &= \sum_{i=1}^n (-1)^i \text{rank}(G_i) \\ &= \sum_{i=1}^n (-1)^i n_i \end{aligned}$$

Then this sum is zero by the exactness of the sequence.

Proof of the lemma:

Call $\text{rank}(\ker(\Phi_i))$ " k_i " and call $\text{rank}(\text{Im}(\Phi_i))$ " I_i ".

Let $k_0 = \text{rank}(\ker(\Phi_0)) = I_1 = 0$ to simplify notation.

$$\begin{aligned} \text{For } i > 0, \text{ each } \text{rank}(G_i) &= \text{rank}(\ker(\Phi_i)) + \text{rank}(\text{Im}(\Phi_i)) \\ &= k_i + I_i \\ &= \text{rank}(\ker(\Phi_i)) + \text{rank}(\ker(\Phi_{i-1})) \\ &= k_i + k_{i-1} \end{aligned}$$

$$\begin{aligned} \text{Thus: } \sum_{i=1}^n (-1)^i \text{rank}(G_i) &= \sum_{i=1}^n (-1)^i n_i \\ &= \sum_{i=1}^n (-1)^i (k_i + k_{i-1}) \end{aligned}$$

But the alternating sum causes all terms to cancel except possibly:

$$k_n + I_1 = \text{rank}(\ker(\Phi_n)) - \text{rank}(\text{Im}(\Phi_1))$$

But by exactness, $\ker(\Phi_n) = \text{Im}(\Phi_{n+1}) = 0$, and $\text{Im}(\Phi_1) = 0$. For consistency, we let $\text{rank}(0) = 0$.

Next we see why rank is useful to know.

5. APPLYING THE RESULTS TO THE OVERLAY PROBLEM

Now let's put some of our results together. We know some homology groups. We have seen one exact sequence, the Mayer-Vietoris Sequence, which relates homology groups for two spaces, their union, and their intersection. Finally we have the *telescoping lemma* which allows us to relate in a single equation the ranks of all of the homology groups that appear in an exact sequence. We merely need to observe how we can actually calculate the ranks of all but one of the homology groups that appear in the Mayer-Vietoris Sequence, and then we will know the remaining group's rank.

Let A and B be two line graphs of maps to be overlaid. Then the portion of the Mayer-Vietoris sequence that may contain non-zero entries is the following:

$$\begin{aligned} \cdots \rightarrow H_2(A \cup B) \rightarrow H_1(A \cap B) \rightarrow H_1(A) \oplus H_1(B) \rightarrow H_1(A \cup B) \rightarrow \\ \rightarrow H_0(A \cap B) \rightarrow H_0(A) \oplus H_0(B) \rightarrow H_0(A \cup B) \rightarrow H_{-1}(A \cap B) \rightarrow \cdots \end{aligned}$$

where both $H_2(A \cup B)$ and $H_{-1}(A \cap B)$ are zero.

The term in the sequence that we want to compute is $H_1(A \cup B)$; and we can find that term by examining $A \cap B$, the intersection graph. Standard graph traversal methods allow us to detect all common components of A and B and to find their intersections. All that remains is to describe $A \cap B$ in terms of its number of disconnected components and its number of independent cycles. Again standard graph traversal techniques permit us to derive these numbers.

Then we know from the Telescoping Lemma that:

$$\begin{aligned} \text{rank}(H_1(A \cap B)) - \text{rank}(H_1(A) \oplus H_1(B)) + \text{rank}(H_1(A \cup B)) - \\ \text{rank}(H_0(A \cap B)) + \text{rank}(H_0(A) \oplus H_0(B)) - \text{rank}(H_0(A \cup B)) = 0. \end{aligned}$$

Furthermore, the rank of a direct sum is just the sum of the ranks:

$$\text{rank}(H_i(A) \oplus H_i(B)) = \text{rank}(H_i(A)) + \text{rank}(H_i(B))$$

Finally, recall that the $\text{rank}(H_1(X))$ is just a count of the interior regions separated by the line graph X ; and $\text{rank}(H_0(X))$ is simply the number of components of X . Putting it all together, and using the notation:

$$r(X) = \text{rank}(H_1(X)) \text{ and } c(X) = \text{rank}(H_0(X)),$$

we get:

$$r(A \cap B) - (r(A) + r(B)) + r(A \cup B) - c(A \cap B) + (c(A) + c(B)) - c(A \cup B) = 0$$

Notice further that if $r'(X)$ represents the total number of regions of the map (not just the interior regions), then the equation still holds (because $r'(X) = r(X) + 1$, and r appears twice with a plus sign and twice with a negative sign):

$$r'(A \cap B) - (r'(A) + r'(B)) + r'(A \cup B) - c(A \cap B) + (c(A) + c(B)) - c(A \cup B) = 0$$

Isolating $r(A \cup B)$ (or r') we get:

$$r(A \cup B) = r(A) + r(B) - c(A \cap B) + c(A \cup B)$$

We conclude with the example in figure 6 to illustrate our methods.

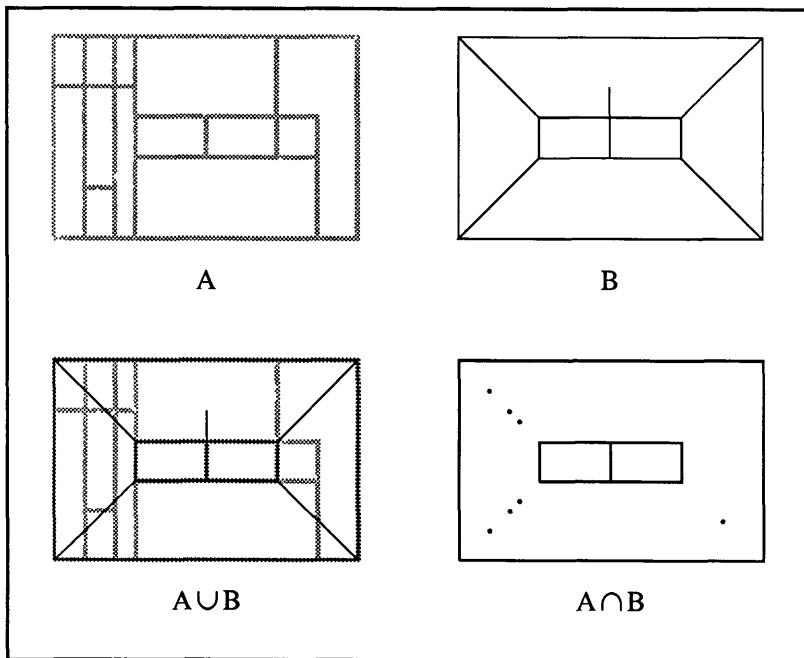


Figure 6: Deriving the Complexity of Overlaying A and B

In figure 6 we see that $r(A) = 13$, $r(B) = 6$, and $r(A \cap B) = 3$. Moreover, because A , B , and $A \cup B$ are all connected, $c(A) = c(B) = c(A \cup B) = 1$. Finally, the number of components of $A \cap B$, $c(A \cap B)$, is 9. Thus by our formula $r(A \cup B) = 24$.

We see from our example that a critical contributor to the sum on the right is the term $c(A \cap B)$, the number of new components (usually isolated intersections) of the intersection graph. By our formula, every new intersection gives rise to a new region! This observation may be useful in estimating the number of new regions that arise in overlay operations. If, for example, we can place a bound on the number of new intersections that will occur, then we can conclude that the number of new regions will be bounded accordingly. This is a nice duality relation that we will develop in a later, longer paper.

6. CONCLUSIONS

We have introduced a few useful ideas from the realm of algebraic topology in order to illustrate one way of applying important duality relations to a specific combinatorial problem. In effect we have converted the problem of determining the number of regions arising from polygon overlay to a graph traversal and intersection detection problem. Further research is planned along the following lines:

1. Describe properties of the line segments in the line networks to be overlaid (such as extent, density, etc.) that would produce a guaranteed bound on the number and type of intersections and a corresponding bound on the number of new regions created.
2. Integrate topological information into the computation of the intersection graph in order to prevent slivers, gaps, and other anomalies due to geometric imprecision.
3. Develop relative homology groups for analysis of local combinatorial duality relationships.

I will write up new results and elaboration of the results sketched here in a more extensive research paper.

7. REFERENCES

1. HENLE, MICHAEL, 1979, *A Combinatorial Introduction to Topology*, W. H. Freeman and Company, San Francisco.
2. HU, SZE-TSEN, 1966, *Homology Theory: A First Course in Algebraic Topology*, Holden-Day, Inc., San Francisco.

PUSHBROOM ALGORITHMS FOR CALCULATING DISTANCES IN RASTER GRIDS

J. Ronald Eastman
Graduate School of Geography
Clark University
Worcester, MA 01610, USA

ABSTRACT

Distance and proximity are critical variables in many geographic analyses. In raster geographic analysis systems, distance is most commonly determined by a sequential growth process whereby distances are accumulated in radial bands from an initial set of features. While such procedures are very efficient for the generation of small buffer zones, they become cumbersome when large distance surfaces need to be determined. As an alternative, two "pushbroom" algorithms are presented -- one for the case of calculating true Euclidian distance over a plane, and a second for incorporating frictional effects in the generation of cost distance surfaces. In the former case, a complete surface of any size can be calculated in exactly four passes through the data. In the second, as few as two complete passes are required, depending upon the nature of the frictional effects encountered. This economy arises from the nature of pushbroom techniques, whereby computations proceed sequentially (not radially) through the raster grid, acquiring directionally-oriented knowledge in accordance with the direction of the pushbroom path.

INTRODUCTION

A common requirement of raster-based Geographic Information Systems is the determination of distance. For example, if the distance from each grid cell to the nearest designated feature can be calculated, a buffer zone of any given distance may then be established. Buffer zones are essential planning tools in the exclusion or confinement of planning activities or investigations. A knowledge of distance is also essential when resources are clustered, and the type or level of activity that may be maintained is consequently distance-dependent. For example, the difference between animal species in the importance of distance to the nearest well is an important consideration in range management (Olsson, 1985, 81). Likewise, when resources are dispersed but access is limited, proximity to access points, such as roads, is an important management concern. Indeed, distance is a common ingredient in the assessment of processes that exhibit distance decay, including processes of mineralization, locational economics, and assessments of risk.

DISTANCE

Two broad approaches to the calculation of distance in raster-based systems are commonly in use. The first, and

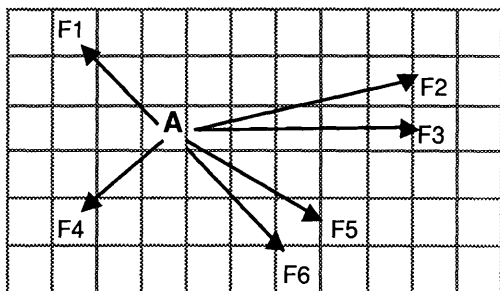


Figure 1 : In the most direct of the distance techniques, distance is calculated as the least distance between a cell and each of the designated feature cells. For example, the distance recorded at position A will be the minimum of the distances between A and each of the feature cells at F1 through F6.

most direct, relies upon simple Pythagorean geometry (Figure 1). Here the distance of each cell to the nearest of a set of designated feature cells is determined by calculating the Euclidian distance from that cell to each feature cell using row and column subscripts (eg., Olsson, 1985, 81). When distances from a single feature cell are required, the technique is quite efficient, with the number of operations being proportional to the square of the maximum distance (in grid cell units) required. In addition, the distances calculated are truly Euclidian. However, as the number of designated feature cells (ie., cells from which distance must be calculated) increases beyond one, the procedure requires that the nearest neighboring feature cell be determined for each cell in the grid. As a result, efficiency is proportional to the number of feature cells as well as the maximum distance involved. In addition, the positions of all cells belonging to the designated features must be known in advance (or be determined from the raster), with their coordinate positions being held in some form of accessible stack or array. Given the complexity of many GIS feature patterns, the technique can thus quickly become bogged down.

To avoid these problems, a second approach (eg., Tomlin, 1986) employs the concept of growth rings. Initially, each of the feature cells is tagged with a distance of 0 while all other cells are marked with a distance equal to the maximum distance that will be determined. Then in a series of passes through the image, distance is "grown" from the feature cells in a series of concentric rectangular "rings" until the maximum distance is reached (Figure 2). The technique has the very strong advantage that the locations of the feature cells do not need to be stored in an accessible list, nor do any nearest-neighbor calculations need to be made. By definition, each growth ring will be constructed with reference to the nearest feature cell, and distance to that feature cell can always be determined by examining the squared distance of an adjacent cell within the previous growth ring. Specifically, squared distance is a linear combination of squared distance in X and squared distance in Y (the Pythagorean theorem). As a result, squared distance is also equal to that of any intermediate distance plus the differences in squared X and squared Y between the new point and that intermediate. Each ring is therefore grown by adding onto the distance of the edge cells the difference in squared X and squared Y. As it

turns out, differences in squared X and squared Y systematically increase by an increment of 2 (Figure 3). The necessary increment can therefore be determined by looking at the previous increment in that direction and adding 2.

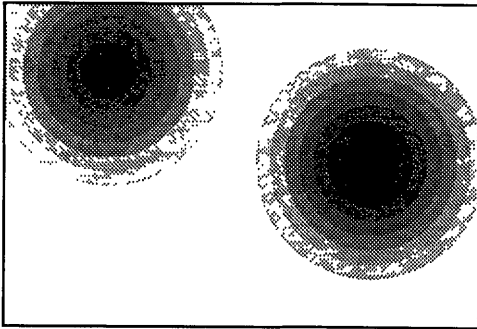


Figure 2 : In a second approach, distance is "grown" in concentric rings around each feature. The shaded bands here indicate the growth ring stages in this process. If the process is continued far enough these rings will coalesce to form a continuous distance surface. Like the method indicated in Figure 1, distances are Euclidian.

	B									
	A	18	13	10	9	10	13	18		
		13	8	5	4	5	8	13		
		10	5	2	1	2	5	10		
		9	4	1	0	1	4	9		
		10	5	2	1	2	5	10		
		13	8	5	4	5	8	13		
		18	13	10	9	10	13	18		

Figure 3 : With distances being stored as squared distances, new distances can be determined by adding incremental changes in delta X squared and delta Y squared. For example, Cell A differs from the cell in the upper-left corner by 0 in squared Y and 7 (ie. $[18-13]+2$) in squared X. The distance of Cell A is thus equal to $18+0+7=25$. Incremental squared distances always differ by 2. Thus the distance of cell B is $18+7+7=32$ -- i.e., the previous difference of 5 in squared X plus 2 plus the previous difference of 5 in squared Y plus 2.

The use of squared distance has several advantages. In addition to being required by the algorithm, the avoidance of square roots significantly speeds operations. Only after all growth rings have been calculated would a final pass be made to take the square roots of cell values. In addition, the intermediate storage of squared distances allows perfect precision with integer data. As a result, rounding errors do not accumulate but only affect the results of the final pass. The procedure, then, does have a considerable amount of appeal. However, it is also not without some pro-

blems. First, if integer arithmetic is to be used, most applications will require 32 bit integers because of the need to store squared distances (with 16 bit integers, the maximum distance that may be calculated is a mere 181 cells). Second, the procedure requires the ability to move quite freely about the image cells. Random access is trivial if the entire image is in memory, but with 4 byte integers or floating point values, the size of image that will readily fit into memory may be quite limited. Random file access can alleviate this, but the speed of random disk operations is typically quite slow. Finally, and perhaps most significantly, the number of passes that must be made through the image is a direct function of the maximum distance that must be calculated. While the determination of narrow buffer zones will be quite efficient, the calculation of a continuous distance surface over any extensive region would likely be quite slow.

COST DISTANCE

An interesting feature of the "growth" procedure is that its logic may also be developed to incorporate frictional effects. Whenever distance is used to imply the cost of movement, that cost will be a function not only of distance, but also of the frictional effects of various relative and absolute barriers such as land cover and slope. This new measure may be called "cost distance", and may be evaluated in any meaningful unit involving distance, money or time.

In the evaluation of cost distance using a growth process (Tomlin, 1986), a matrix is first constructed containing the designated feature cells marked with a distance of 0, and with all other cells being tagged as unknown. In addition, a second matrix is formed in which the frictional effect of each cell is stored. All frictions are indicated with a value relative to 1. Thus, for example, a friction of 2 would indicate that it costs twice as much as usual to pass through that cell. The procedure then involves a series of passes through the matrix in which unknown cells which are adjacent to a cell of known distance are given a distance equal to the known cell plus one times the frictional effect in the cardinal directions and a distance equal to the known cell plus 1.41 (square root of 2) times the frictional effect in the diagonal directions (Figure 4).

Feature image	+	Friction image	=	Cost distance																											
<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>???</td><td>???</td><td>???</td></tr> <tr><td>???</td><td>0</td><td>???</td></tr> <tr><td>???</td><td>???</td><td>???</td></tr> </table>	???	???	???	???	0	???	???	???	???	+	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>1.00</td><td>1.00</td><td>1.00</td></tr> <tr><td>1.00</td><td>1.00</td><td>2.00</td></tr> <tr><td>2.00</td><td>3.00</td><td>3.00</td></tr> </table>	1.00	1.00	1.00	1.00	1.00	2.00	2.00	3.00	3.00	=	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>1.41</td><td>1.00</td><td>1.41</td></tr> <tr><td>1.00</td><td>0.00</td><td>2.00</td></tr> <tr><td>2.82</td><td>3.00</td><td>4.23</td></tr> </table>	1.41	1.00	1.41	1.00	0.00	2.00	2.82	3.00	4.23
???	???	???																													
???	0	???																													
???	???	???																													
1.00	1.00	1.00																													
1.00	1.00	2.00																													
2.00	3.00	3.00																													
1.41	1.00	1.41																													
1.00	0.00	2.00																													
2.82	3.00	4.23																													

Figure 4 : Distances grown outwards from any feature cell would normally result in an increase of one in the cardinal directions and 1.41 along the diagonals. However, frictional values other than 1.00 will proportionately alter this relationship.

Unlike the simple distance growth model, the concentric "rings" are no longer rectangular in shape, but octagonal. In addition, distances are accumulated directly, rather than as squared distances. The reason for this relates to the fact that there is no longer any predictable relationship between distance and the difference in X or Y between a grid cell and its nearest target (because of the variable effects of friction). Errors will therefore accumulate for any cost distances determined along paths other than one of the cardinal directions or the diagonals. As a result, most systems, such as IDRISI (Eastman, 1987) and the Map Analysis Package (Tomlin, 1986) provide both a simple distance routine as well as one for calculating cost.

Given its inherent "growth" logic the cost distance routine discussed above shares most of the same strengths and weaknesses as the simple distance growth routine. Again, the procedure is very efficient whenever cost needs to be determined over a narrow buffer zone. However, it likewise bogs down whenever a significant region must be determined. For example, to calculate a continuous cost surface over a 512 by 512 grid could involve over 700 passes through the data set to construct the required number of growth rings. Similarly, the need for random access can cause a tradeoff between image size and speed.

THE IDRISI SYSTEM APPROACH

During the development of the IDRISI GIS system, new procedures for the calculation of distance and cost surfaces were developed. The IDRISI system is a grid-based (or "raster") geographic analysis system that has been developed by the author at Clark University. It is also distributed by the university with over 600 registered users at this time.

The IDRISI system was specifically designed to operate in a microcomputer environment in which disk space is plentiful (eg. 32 Mb), but random access memory is scarce (640 Kb). As a result, all procedures were developed using a scan line approach whereby only a limited number of scan lines would be operated upon at one time. The procedures developed for the calculation of distance and cost thus follow a scan-line approach whereby successive rows of the image are read and operated upon, and then saved back to disk. In both cases, the procedures operate by pushing effects through the image, much like a pushbroom would be used to systematically clean a room. Effects then ripple through the image, much like water being pushed over a wet floor.

THE PUSHBROOM DISTANCE PROCEDURE

In the case of the simple Euclidian distance algorithm (called "DISTANCE" in IDRISI), processing starts from the upper-left cell and proceeds along each row and then sequentially down the image from one row to the next. This is identical to the order in which the image is stored. As the feature image (from which distances must be calculated) is read, a temporary data file is output with records which record the distance in X and the distance in Y (as floating point real numbers) to the nearest target cell that is ei-

ther above or behind it. If no feature has yet been found, these values are output with a special flag value. However, once a feature is found, distance in X and Y are carried along by incrementing by one for each successive column or row. When more than one feature has been found, delta X and delta Y are recorded from the nearest one by comparing distances determined from incrementing the delta X and delta Y values from the cell to the left, the cell diagonally above, and the cell directly above. In essence, the effect is one of determining the lower-right quadrant of all growth rings in a single pass --a kind of ripple effect in which knowledge of feature positions is carried along in the pass (Figure 5).

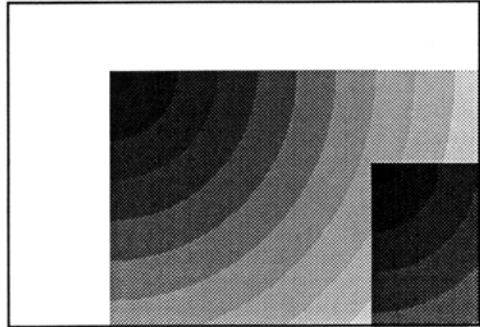


Figure 5 : During each pass of the pushbroom procedures, effects are "pushed" from regions already examined (in this example, from cells above and behind) into regions yet to be processed. The procedure thus attains directionally-oriented knowledge with each pass.

This procedure is then repeated, three further times. Whereas the first pass proceeded from top-left to bottom-right, successive passes then proceed from bottom-right to top-left (to determine the upper-left quadrant of the growth rings), from top-right to bottom-left (to determine the lower-left quadrant) and finally from bottom-left to top-right (to determine the upper-right quadrant). These four passes are then overlaid with the final output being stored as the minimum distance calculated from the delta X and delta Y figures for each pass.

In its implementation, several economies can be used. First, as each row is processed, only that row and its immediately preceding row need be held in memory. By using scan-line buffers in memory, the beginning of any required row (regardless of the direction in which it would be processed) would be randomly accessed, with all remaining row values being read sequentially in normal file order. Second, the overlay step does not need to be done at the end, but can be done on a row by row basis during each pass. All that is required is that delta X and delta Y be recorded in a consistent coordinate system with negative values in the left and bottom quadrants. In this way, the results stored after each pass represent the best estimates of least-distance delta X and delta Y for all passes up to that point. Finally, all intermediate distance calculations can be made

using squared distance. Only on the last pass does the square root need to be taken.

As a result of this procedure, distance can be calculated as a continuous surface in four passes regardless of the size of the image, the number of feature cells, or the maximum distance required. That said, it does require that the full surface be calculated every time. To create a buffer zone, then, requires that the distance surface be reclassified into cells within the zone distance and those outside it. One might expect, then, that the procedure would be slower than the traditional growth ring approach for narrow buffer zones but faster whenever a more significant region must be defined.

THE PUSHBROOM COST PROCEDURE

In the case of the cost distance routine (called "COST" in IDRISI) a somewhat similar procedure is used. Again, sequential passes are made through the data, but as few as two complete cycles are required depending upon the nature of the frictional effects involved. First, however, the issue of friction needs to be discussed.

Frictional effects present barriers that are either absolute or relative in nature. An absolute barrier is one in which the frictional effects are so high that movement cannot proceed through that cell. Relative barriers, however, do allow movement, albeit at an additional cost. Some systems treat absolute barriers as special cases. However, in IDRISI, an absolute barrier is indicated simply by giving that cell a friction that is impossibly high (ie., one that will always cause distance to be shorter by moving around the barrier than over it).

Given this concept of friction, the cost distance procedure processes the file from top-left to bottom-right and then backwards from bottom-right to top-left. During the first pass it sets all unknown cells to have an extremely high distance, and all feature cells to have a distance of zero. Additionally, like all subsequent passes, it examines the 8 neighbors about each cell to see if distance incremented from that neighbor is less than the distance currently stored for that cell. Like the growth procedure for calculating cost, distance is incremented as one times the friction in the cardinal directions and 1.41 (square root of two) times the friction along the diagonals.

As long as there are no absolute barriers (ie., as long as going over a feature is always less expensive than going around it), the complete cost surface can be determined in two full passes from the position of the first feature. For example, if the first feature cell is found half-way through the image, the procedure would minimally require the first pass down the image (in which the first feature is found), the second pass back up the image, and a third pass back down the image until the position of the first feature cell is found again. When absolute barriers are present, however, their nature and position may disturb this rule. For example the barrier in Figure 6 Part A, causes no problems since information from the feature is carried to all parts of the image. In Figure 6 Part B, however, the position of the barrier prevents information about the location of the feature from being carried to the bot-

tom right-hand corner on the first pass. Unless a complete third pass is undertaken, distances will not be correct in the region indicated. Generally, the procedure will have difficulty with absolute barriers that produce maze-like corridors. However, for natural resource applications where relative barriers predominate and absolute barriers are not complex, three complete passes have generally been found to be adequate.

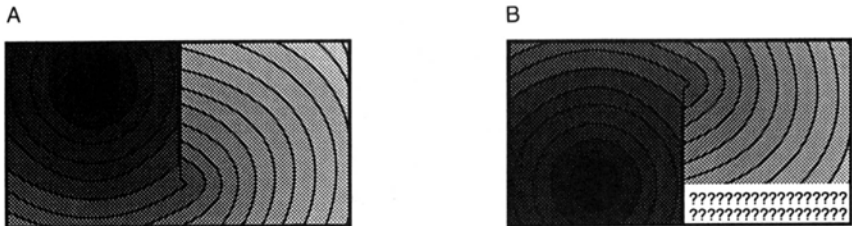


Figure 6 : Absolute barriers may necessitate more passes than the minimum required for a surface with only relative barriers. The absolute barrier in A, for example, poses no problems for the routine since the top-left to bottom-right and vice-versa pass orientation will carry information about the position of that feature to all cells in the image. The barrier in B, however, will block the complete spread of information unless at least three complete passes are used.

As with the pushbroom procedure for calculating simple distance, the pushbroom cost procedure produces an algorithm where the number of passes is independent of the maximum cost distance to be determined. Similarly, since the entire image is processed, the procedure is quite efficient in instances where a complete cost surface is required.

A COMPARISON OF TECHNIQUES

A comparison of the pushbroom and growth ring techniques is difficult to evaluate. The strengths of one are the weakness of the other. For the determination of small buffer zones, there is little doubt that the growth ring procedures will be superior, since their speed is directly related to the size of the buffer required. However, whenever a full distance surface is required (cost or simple Euclidian), the pushbroom techniques should be faster. To evaluate this, the two techniques were compared by applying them to identical full-surface problems. The first test (to be called the "center" test) involved a single-cell feature in the center of the image while the second (to be called the "corner" test) involved an image with single-cell features in each of the four corners of the image. Growth procedures typically limit their operation to the maximum sub-region required for processing during any one cycle. As a result, operations should be fastest for the "center" test and worst for the "corner" test. For the pushbroom procedures, however, the results of these two tests will be

identical. These two tests were then applied to both the simple Euclidian and cost distance problems for images which ranged in size from 25 x 25 cells to 175 x 175 cells. For all cost distance tests, frictions were set at 1.0 for all cells.

For the growth procedures, the "SPREAD" routine from the microcomputer version of the Map Analysis Package

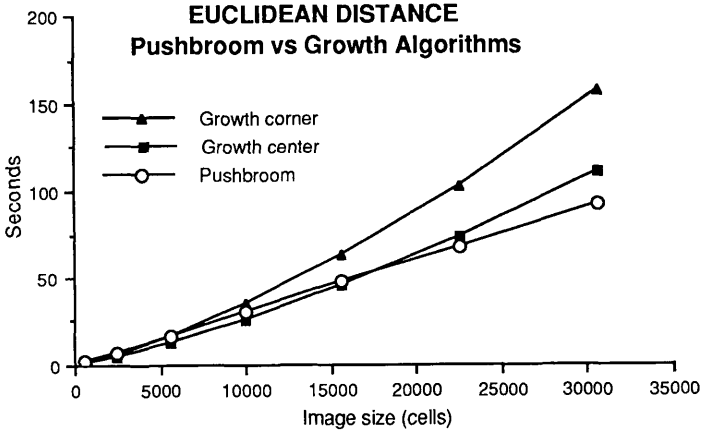


Figure 7

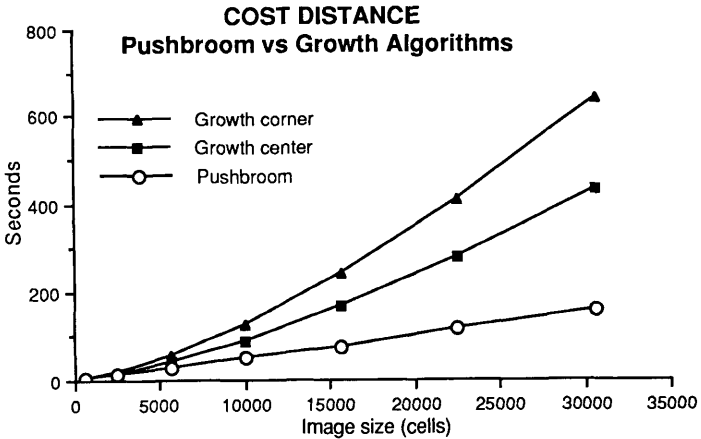


Figure 8

(Tomlin, 1986) was used. This routine incorporates both the simple distance and cost distance growth routines (which it switches between depending upon whether one specifies a friction surface to be spread "through"). For the pushbroom routines, the "DISTANCE" and "COST" modules from the math coprocessor version of IDRISI were used (a version still under development as of this writing). Both systems were run on a 20 Mhz 80386 computer with a 128 Kb disk cache in use. In addition, both programs accessed the 20 Mhz 80387 coprocessor installed on the test machine. Figures 7 and 8 present the results of these two tests.

As can be seen from these results, the times for the pushbroom algorithms are a linear function of image size. For the growth routines, however, processing time can be seen to be exponentially related to image size with times for the corner test being, as expected, uniformly greater than for the center test. The images used in these tests were not large, and yet the savings in processing time afforded by the pushbroom techniques are immediately apparent -- particularly for the cost distance tests. For example, extrapolating these results to a 1024 x 1024 image (this test could not be done with the Map Analysis Package since its maximum image size is 32,640 cells [180 x 180]), the pushbroom cost procedure would require 1.5 hours while the growth ring procedure would require almost 11.5 hours for the center test and 20 hours for the corner test!

CONCLUSIONS

From the above, several broad conclusions can be reached about the techniques introduced in this paper. First, the pushbroom procedures provide a logic compatible with scan-line processing. As a result, they may be applied to very large images even though memory may be limited (such as the 640 Kb address space of MS-DOS). Second, like growth ring procedures, they do not require explicit information about the location of the features from which distance is to be determined. Third, their speed is a linear function of image size. Fourth, they operate upon the entire image, thus making them somewhat inefficient for determination of small buffer zones (for which the growth ring procedures excel). But finally, they are substantially faster than the growth ring processes whenever a more substantial buffer zone must be processed or a continuous distance or cost surface must be determined.

REFERENCES

Eastman, J.R., (1987) IDRISI, A Grid-Based Geographic Analysis System, Graduate School of Geography, Clark University, Worcester, MA, USA.

Olsson, L. (1985) An Integrated Study of Desertification: Lund Studies in Geography, Ser.C, No.13.

Tomlin, C.D., (1986) The IBM Personal Computer Version of the Map Analysis Package, The Laboratory for Computer Graphics and Spatial Analysis, Harvard University, Cambridge, MA, USA.

SPATIAL ADJACENCY - A GENERAL APPROACH

Christopher M. Gold
Department of Geography
Memorial University of Newfoundland
St. John's, Nfld., Canada A1B 3X9.
BITNET Address: CGOLD@MUN.

ABSTRACT

It is fact universally acknowledged that discrete computing systems are ill-equipped to process vector-based spatial information: inexact line intersection calculations and similar geometric (co-ordinate) operations can not readily guarantee consistent graphical structures (topology). It is proposed here that use of Voronoi diagrams, especially euclidean-distance nearest-object Voronoi diagrams of points and line segments in the plane, permits a general-purpose conversion of geometric information to a graphically-structured form amenable thereafter to graph traversal and other fundamental discrete operations appropriate to the computing environment employed. While the divide-and-conquer approach is efficient, object-at-a-time insertion and deletion techniques build on the current adjacency structure; preserve it and are consistent with database updating methodology; and direct comparisons can be used between one and two-dimensional linked-list operations. This approach permits the handling of spatial information in a manner consistent with computer strengths - by using linked-list, graph-traversal and tree-search algorithms well known to computing science to answer a wide variety of basic geographic queries, including interpolation, spatial ordering and medial-axis transforms.

INTRODUCTION

This paper is intended to review the applications of Voronoi diagrams to a wide variety of spatial adjacency problems, with particular emphasis on applications in automated cartography and geographic information systems. Topics covered include: what is a Voronoi diagram? What is the current research in this field? What can they do for us? How may they be implemented - firstly involving the implementation of general polygon structures and their duals, and secondly referring explicitly to Voronoi polygons rather than general polygons? Reference is made to the significance of boundaries in these general polygon structures, and then to the construction techniques for Voronoi diagrams. A comparison is made between the construction of general two dimensional triangular networks and the more conventional one dimensional linked-lists and trees familiar to computing science. Having provided a general background, discussion then covers a variety of applications, including interpolation, skeleton encoding, and spatial ordering.

WHAT ARE VORONOI DIAGRAMS?

Consider a set of objects (points) in the plane. Each of these objects may be considered to have a sphere of influence, defined as the region which is closer to that object than to any other object. The result of this zoning activity is to partition the plane into a set of polygonal regions, each region associated with a particular object. For points in the plane these polygonal regions can be shown to be convex polygons. The result of this process is referred to as a Voronoi tessellation.

While the mathematical definition is straightforward it must be emphasized that Voronoi diagrams are not at all abstract entities. They may be created by the use of blotting-paper and wicks, the magnetic fields of adjacent magnets, etc. (see Morgan, 1967). Thus Voronoi diagrams are closely related to real physical processes, which simplifies both the visualization of the technique and the potential for the modelling of these physical processes.

Considerable research activity has been dedicated to studying Voronoi diagrams in the last few years. While theoretical algorithms are the particular specialty of the field of computational geometry, the applications aspects have not yet been fully explored. The efficient construction of point Voronoi diagrams in the euclidean plane has been well known for some years, but other particular Voronoi diagrams - using other metrics, furthest-point Voronoi techniques, cases with boundaries, and other special applications - are still subjects of ongoing research. The major sources of information on the topic are the textbook by Preparata and Shamos (1985), and the ACM/SIGGRAPH annual proceedings on computational geometry. The approach has various characteristics, which include the use of "divide and conquer" methods to obtain the most efficient construction techniques. These result in methods that are not necessarily the easiest to implement on a computer, and in many cases have not been implemented. Finally, the use of divide and conquer techniques implies the construction of the diagram for the whole data set at one time, rather than permitting the updating of the data set in the process of the application.

VORONOI DIAGRAMS AND CO-ORDINATE GEOMETRY PROBLEMS.

Problems in co-ordinate geometry arise frequently in computer implementations of a variety of science and engineering applications. These are associated with the fact that the specification of geometric x,y co-ordinates for some object being described does not automatically provide information about the relationships between line segments or objects themselves. Thus in both automated cartography and computer aided design the specification of object co-ordinates is not sufficient to link these defined objects together to form a coherent whole. As a general statement, co-ordinates do not of themselves produce relationships, that is: graph theoretical structures relating objects in space. This is due partly to the finite resolution of computer word lengths representing co-ordinates of intersection points etc., but

primarily because the two branches of mathematics involved have very little overlap in the problems described here. Graph theoretic techniques require that relationships (adjacency relationships in particular) be previously defined, while the straightforward definition of co-ordinates in conventional geometry provide no information of itself about the linkage between points and objects in space.

It is suggested in this paper that the use of a Voronoi generating process may simplify the transition from co-ordinate based information to graph theoretic (adjacency) based structures. Once graph theoretic structures are available many otherwise difficult processes may readily be implemented on the discrete machines available for computing problems. The rest of this paper will therefore be concerned with the storage of polygons in a computing environment, the specific issues of creation and storage of Voronoi polygons in the computer, and applications that ensue from the availability of the resulting structures.

THE STORAGE OF GENERAL POLYGON INFORMATION

Given any map composed of polygons, several things should be noted. Firstly, the two dimensional plane is entirely covered by adjacent polygons: there are no gaps. Thus every polygon has an adjacent polygon, with special care being taken at the boundaries of the map. Secondly, in the two dimensional plane there are only three basic classes of objects: points or nodes (zero-), arcs (one-) and polygons (two-) dimensional objects. Thus a polygon may be defined by its several boundaries, by its several nodes, which occur at the junctions between boundaries, and also by the several adjacent polygons that bound it. Line segments or arcs may be defined in terms of the two end points (nodes), and also the "left polygon" and "right polygon". Information about nodes could include all of the arcs or boundary segments that meet at it and in addition all of the polygons that themselves meet at that node. A useful summary of the options for storing the relationships between polygons, arcs and nodes may be found in Gold (1988a).

A polygon set is in fact a graph. A graph is formed of regions, edges, and nodes, which are directly related to the polygons, arcs and nodes previously discussed. Graph nodes have a valence associated with them - that is, the number of edges that meet at that node. In a two dimensional planar graph, such as a map, most nodes will have a valence of 3. All nodes with a valence of 4 or more may be reduced to nodes of valence 3 by inserting dummy line segments of zero or near-zero length into the data structure. Thus if we can restrict ourselves to nodes of valence 3, all polygons may be represented by the dual triangulation. The dual of a graph is formed by replacing all regions (polygons) with nodes; replacing all nodes with regions; and replacing all edges (boundaries between adjacent polygons) with new edges that connect the "centres" of each original region. Thus polygons convert to nodes, nodes convert to triangles (since they are all of valence 3) and edges convert to new edges.

Figure 1 shows a polygon set and the associated dual triangulation. Polygons A through F are represented in the dual by nodes A through F. Each triangle edge represents or shows the existence of an original polygon edge, and any property associated with that original polygon-polygon boundary may now be associated with the new triangle edge. Thus in a computer structure the triangle edge may point to the x,y co-ordinates forming the irregular polygon-polygon boundary and may also inform the user of the kind of boundary involved. It should be noted that the boundaries need not be simply hard lines as is conventionally represented on a map, but may involve other properties such as fuzziness, faintness, convolutedness, or even flow between adjacent polygons. Thus a boundary - and here a triangle edge - represents a relationship between two adjacent polygons. This relationship may be of any type required by the application. Thus if a soil type map is known to have gradational boundaries between soil types, as is usually the case, and if the soil scientist can describe this gradational relationship, the data structure is capable of preserving this information for future use.

Thus a triangulation structure permits the storage of information concerning polygons, arcs and nodes. A triangulation is one appropriate data structure, since in the two dimensional plane nodes are usually of valence 3. Thus a triangulation network is a relationship storage device. One of the advantages of preserving triangulations rather than polygon sets in the original form is that triangulations have a known number of vertices and edges, simplifying internal storage concerns in a computing system. One possible way of storing a triangulation data structure is to preserve the three adjacent triangles and the three vertices for each triangle record (see Gold et al., 1977). In that particular case triangle edges are not themselves preserved. Another alternative is to preserve the triangulation as a series of edges rather than as a series of triangles: each edge record consists of a "from" node , a "to" node and the next edge record clockwise (or anti-clockwise) from each end node. This particular data structure is also of fixed length, and hence of simple implementation, but in addition detailed information about the boundary itself between any two polygons may readily be added. See Gold (1988a) for more details on the selection of data structures. While both of these data structures, as well as variants, are appropriate formats for the storage of the dual triangulation of a polygon set, the line record format appears to be better where arbitrary boundaries are involved, whereas the triangle record format, while not preserving any specific boundary information, seems to be particularly appropriate to the storage of Voronoi diagrams, where polygon boundaries are not arbitrary but are implicit in the relationship between the two adjacent map objects. As will be seen, in the preservation of Voronoi polygons in a computer data structure, the storage of the junction between the three boundary segments is the most useful property to preserve, and one of these "circumcentres" exists for each triangle.

THE IMPLEMENTATION AND STORAGE OF VORONOI POLYGONS

We have discussed the storage of general polygons and some of the possible data structures to use. As previously mentioned, the triangulation data structure appears appropriate for the preservation of Voronoi polygons in particular. Figure 2a shows a simple set of points in the plane, their associated Voronoi polygons (solid line), and the resulting dual triangulation (dashed line). In this particular case, rather than using divide and conquer techniques to generate the whole Voronoi diagram at once, individual points are inserted one at a time into the data structure. In Figure 2a, a new data point, marked X, is to be inserted into the data set. Figure 2b shows the results of inserting the new point and in consequence creating a new polygon at the expense of the previously existing polygons. Figure 2c shows the portions of the previous polygons "stolen" by the new Voronoi polygon. This simple insertion technique is theoretically less efficient than divide and conquer methods, but it is simple to implement and cost-effective for all but the largest data sets. In addition, the ability to insert and delete individual points is crucial in many applications. Figure 3 shows the result of generating the Voronoi polygons, and dual triangulation, for a test data set from Davis (1973).

The objects inserted into the two dimensional plane need not be restricted to points. Figure 4 shows the case where individual points and line segments are inserted. Some increased complexity therefore exists - in particular, while the boundary equidistant between two adjacent points is a straight line, and the boundary between two adjacent line segments is also a straight line, the boundary between a point and a line segment forms a parabola.

In order to insert a line segment into a Voronoi diagram, first of all the two end points must be inserted as described previously, and then the connecting line segment itself added. This is consistent with the fact that connecting the two end points adds additional information to the map. The Voronoi region for a line segment therefore has boundaries that consist of straight line segments and parabolic segments, and it need not necessarily be convex. Figure 5 illustrates the insertion of a line segment into the Voronoi diagram.

Figures 4 and 5 also show the triangulation of the Voronoi regions. Point objects are represented as solid dots and line objects are represented as dashed lines. Line segments are considered as distinct objects from their end points. Since the Voronoi regions are in fact polygons the result is a triangulation as described previously for general polygons. Since the Voronoi regions are entirely defined by the relationships between points and points, points and lines, or lines and lines, there is no need to save explicit boundary information and thus no need to implement the line segment data structure previously described. On the other hand, the junctions between line segments and parabolic segments are of considerable importance, and as one of these junctions is associated with each triangle, a triangulation

based data structure appears appropriate for this problem. In a simple point-Voronoi diagram the junction of these three boundaries is at the circumcentre of the particular triangle. When the Voronoi diagram is extended to include line segments a similar definition holds: the centre is at an equal distance from the three objects at the vertices of the triangle. Thus an appropriate circle would pass through any vertex that consisted of a data point, and would be tangent to any vertex consisting of a line segment. Nevertheless, all triangles in this structure have a circle centre and radius representing the maximum distance one can get away from each of the three vertex objects. For further details see Gold (1988c).

In conclusion boundaries are implicit between objects of known type, therefore Voronoi boundaries need not be explicitly preserved. The intersections of Voronoi boundaries define the available valid relationships. Thus the dual triangulation data structure with "circumcentres" should be preserved to define adjacency relationships based on euclidean distance.

RELATIONSHIPS BETWEEN GENERAL POLYGONS, DUAL TRIANGULATIONS AND ONE DIMENSIONAL LINKED LISTS.

Fundamental operations on one dimensional conventional linked lists include the following basic operations. Firstly: an initialize process, usually involving setting up two end nodes with values selected to be outside the range of the data to be inserted. Secondly: an insert operation, permitting the insertion of a new node between two previous nodes. These nodes, in a linked list application such as simple sorting, would each consist of a left pointer, a right pointer, and a value field - probably containing one of the numeric values to be sorted. Assuming that the linked list is to be maintained in ascending numerical order, a search technique must be available to determine whether a particular numerical value has either already been inserted, or alternatively to determine the values immediately below and immediately above the new value to be inserted. This search algorithm could involve either a simple "start at one end and keep looking until you get there" process, or a more elaborate binary search. A third necessary linked-list operation would be a delete procedure, permitting the deletion of a particular value no longer desired, and the elimination of the associated node in the linked list. Finally, in some cases (e.g. a bubble sort) a "switch" operation may be of value. This operation switches the values of two adjacent nodes. All of these operations, with the exception of the search, are of $O(n)$ efficiency. The efficiency of the search technique itself may vary from $O(n^2)$ for a simple minded "read the whole list", to $O(n \log n)$ for either a binary search technique or else a tree search - if it has been considered desirable to include a hierarchical tree structure above the one dimensional linked list.

In the case of a set of general polygons (not specifically Voronoi) we can create an equivalent set of operations. An initialization operation consists of defining a large

exterior polygon, such as a map boundary, enclosing all subsequent data. This region will be divided into a space-covering polygon set as data is inserted or deleted. A partially-completed polygon set is shown in Figure 6a. The dual triangulation is also illustrated. Note that each node in the dual triangulation represents one of the original polygons, and each triangle in the dual triangulation has one associated node (with valence 3) in the original polygon diagram.

In Figure 6b the central polygon has been divided into two by a new boundary. The result of this operation is to create one new boundary segment and two new 3-valence nodes. Thus in the dual triangulation representation two new triangles have been created. This "split" process may be replaced by a reverse "merge" process. In this case a boundary between two adjacent polygons is deleted, and hence two polygons become one. In the dual triangulation representation two adjacent triangles are deleted, and the two nodes on their common boundary are merged into one.

We may therefore consider the equivalent of a simple insert process in a one dimensional linked list to be a split process in the two dimensional polygon context. Thus rather than "inserting" a new node we are splitting one node into two. This is appropriate since in the polygon problem it is assumed that the whole plane is tiled in polygons. The equivalent of a one dimensional delete process is the merge process described above for the polygon problem. Thus for any general polygon set we have the equivalent of insertion and deletion in a one dimensional linked list. In addition, this is readily implemented in the dual triangulation of the space-covering polygon set.

An additional property of this insert/split approach is that it allows us to subdivide space in a hierarchical tree fashion without imposing any specific restrictions on the shape of any particular set of polygons. Thus the insert (or split) process involves the taking of the initial polygon, let us call it AB, and splitting it into two sub-polygons A and B. In terms of conventional tree structures this produces a binary tree with all polygons at the leaves. The delete/merge process takes two leaf polygons A and B, deletes them both and replaces them with their common parent polygon AB, which itself becomes a leaf.

The tree structure previously suggested is directly relevant to problems concerning the order of efficiency of the search process. The simplest one dimensional search technique is merely to "walk" through the linked list starting at one end until the appropriate value in the ordered list is found. For multiple searches it is reasonable to continue the new search from the point of termination of the previous one. This local walk technique can be applied to a triangulation in two dimensions. For details see Gold et al. (1977). This walk through a triangulation in two dimensions is approximately of $O(n^{1.5})$, as opposed to $O(n^2)$ for the one dimensional case. The walk in two dimensions is based on geometric criteria - thus it is readily used in the case of Voronoi polygons and dual triangulations, where the geometric

relation between the triangulation and the dual polygons is straightforward, but the approach is less obvious where the dual triangulation is of a general polygon set, the boundaries are arbitrary and it is unclear where the appropriate "centres" of the original polygons should be.

Nevertheless for the Voronoi polygons a simple geometric walk is readily implemented and reasonably efficient under most circumstances, since data on input is usually naturally ordered by the process of acquiring the data in the first place: thus there is a tendency for the next data point to be inserted into the data structure to be close to the previous one. Where a higher order of efficiency is desired the binary tree structure previously mentioned may be implemented. Note that no rules have been given as to precisely when two polygons should be split or merged. This would be a function of the particular mapping information desired. It is therefore flexible, but does require implementation of splitting and merging rules based on knowledge of the data. It is nevertheless the same technique - whether applied to simple hierarchical subdivision by map sheet, subdivision by census district, county and higher order region, or any other desired natural hierarchical order to the polygon data.

The last of the processes to be described is the "switch" operation. Any two adjacent triangles will have a common boundary. The quadrilateral formed by these two triangles may be divided into two triangles either in the original fashion or by connecting the two opposing points - thus changing the diagonal of the quadrilateral. This was previously described in Gold et al. (1977). The switch operation is equivalent to the switching of two nodes in a one dimensional linked list. However, in order to decide whether a pair of triangles should be switched in any particular case, an appropriate criterion should be used. The most appropriate criterion is generally accepted to be the Voronoi. On this basis triangles perturbed by nearby network modifications may be tested to see if an adjustment (switch) is required to preserve the Voronoi property. Thus the testing and switching of all edges of the triangulation that have been modified by insertion or deletion, or by the switching of nearby edges, permits the ready preservation of the Voronoi property for any object insertion or deletion. This operation can be guaranteed to be a local process - in fact on the average the insertion of a new data point can be expected to cause 6 switch operations to be performed. Thus no insertion or deletion in one corner of a map sheet can have any influence on remote portions of the triangulation.

We have thus shown for the case of the general triangulation the relationships that exist between the basic operations of initialize, insert, delete, switch and search in the one dimensional linked list case well known to computer science, and the two dimensional triangulation case which may be applied to any space-covering polygonal set. In the special case of the Voronoi polygons the switch operation can maintain the Voronoi criterion subsequent to any perturbation of the network by insertion or deletion processes.

APPLICATIONS

The primary function of the implementation of the Voronoi tessellation for a set of points or line segments is to allow coordinate geometry problems to be approached from the graph theoretic viewpoint. Some specific applications are given.

Figure 7 is taken from Gold and Cormack (1987). The ordering techniques were first described in Gold et al. (1977). If a triangulation has been formed by the previously mentioned techniques (not necessarily Voronoi) it is possible to perform operations upon triangular elements of this network in a spatially consistent order. In the example shown a viewpoint labelled X is located near the centre of the triangulated data set. After the first triangle has been processed there remain three adjacent triangles. Each of these may be processed in turn. These subsequent triangles have either one or two adjacent triangles that are further away from the viewpoint than they are themselves. By appropriate geometric tests, described in Gold and Cormack (1987), it is possible to process each triangular element in a nearest-to-furthest order with respect to the specified viewpoint. Thus, since the triangulation may be ordered, so also may the objects from which the Voronoi polygons, and the dual triangulation, were produced. This permits the general solution of a variety of adjacency problems. Hidden line problems may be processed in a front to back or back to front ordering with respect to the eye position by following this procedure. For pen plotter applications pen movement may be minimized by processing the map in an order based on the triangular patches formed by the triangulation process. Radial searches outwards from the viewpoint are readily performed using the technique, permitting easy retrieval of all data objects close to the desired starting location. This graph theoretic approach is particularly desirable where a selection of neighbours is required, as in interpolation.

In interpolation problems, such as traditional contouring or perspective view modelling, it is difficult to generate an interpolated surface that will always honour every data point, whatever their distribution. Figure 2a shows a simple Voronoi tessellation of a small point data set. Figure 2b shows the result of inserting a new data point, marked X. This new point however is not a "real" data point, but simply a sampling location where an elevation value is desired. Figure 2b shows the new polygonal region carved out from the Voronoi polygons of the real data points themselves. Figure 2c shows the areas of each of these polygons "stolen" by the Voronoi polygon of this new dummy point. These stolen regions are of considerable interest, as they permit straightforward interpolation between arbitrarily distributed data points. The relative areas stolen from adjacent data points are used as weighting functions to generate a weighted average of these adjacent points, to form the estimated elevation at the point marked X. A particular strength of this approach is that only neighbouring data points which have a finite positive area stolen from them are defined as neighbours to the interpolation point X. Thus no discrepancy exists between the selection of the neighbouring points and the weighting function used upon them (see Gold, 1988b, 1989).

As an additional application, Figure 8 shows a map of a small village region. A variety of roads, houses, streams etc. are displayed. In any geographic information system it is frequently desirable to be able to determine which map entities are adjacent to which other map entities. An example would be to determine which houses are adjacent to a particular road. It is of course possible to generate Voronoi zones about each object defined on the map. First of all it is necessary to break up certain features such as roads into individual segments - this is a cartographic problem not addressed in this paper. The result of constructing the Voronoi diagram of the major objects on this map is also shown. On the basis of this Voronoi diagram it is possible to make reasonable statements about whether a particular house, shed, etc., is adjacent to a particular road, or to another building. The answer to this question would be "yes" if the Voronoi regions of the two objects under query are adjacent to each other and have a common boundary. Indeed the extent of the common boundary between them could be a measure of the adjacency itself. Note that in a few cases, e.g. where a stream goes under a road, nodes with an order of 4 as opposed to an order of 3 may be found on the Voronoi diagram. As before, this Voronoi diagram can be expressed as a dual triangulation. For details see Gold (1987).

A final application concerns the skeleton encoding of polygons. Figure 9a shows a polygon with one concave vertex. A "wave-front" analogy has been used to show the growth inwards of parallel bands along the boundary itself. Figure 9b shows the result when these wave-fronts have met and completely engulfed the original polygon. Each line segment on the original boundary now has associated with it an interior region bounded by edges formed where the various wave-fronts met. These regions are the interior components of the Voronoi region for each of the line segments of the boundary (and as such have a valid dual triangulation). In the case of the single concave vertex shown, an interior region is defined for the vertex itself, and not merely for the line segments involved. (Figures 4 and 5 illustrate point and line Voronoi diagram generation.) This interior boundary between a convex vertex and the opposing line segment generates a parabolic interior segment to the polygon skeleton. (This example is taken from Brassel and Jones, (1984), where "bisector skeletons" perform a similar operation.) This polygonal skeleton is of value as a graph theoretic description of the general shape of the polygon, and as such (in raster mode) is frequently used in character recognition applications. In the field of cartography the technique is of value as a label or name placement aid.

CONCLUSIONS

It is hoped that this paper has shown the basic relevance of the Voronoi tessellation as an aid in converting co-ordinate geometric problems to graph theoretic approaches. On this basis a large variety of applications may be attacked using a common set of tools. The basics of the approach have been described along with appropriate data structures, and several applications have been outlined. Other applications are expected to be developed in the near future.

ACKNOWLEDGEMENTS

The funding for this research was provided in part by an operating grant from the Natural Sciences and Engineering Research Council of Canada, and in part from the Energy, Mines and Resources Canada Research Agreement Programme.

REFERENCES

Brassel, K.E. and P.L. Jones, 1984, The construction of bisector skeletons for polygonal networks. IN: Proceedings, International Symposium on Spatial Data Handling, Zurich 1984, v.1 pp. 117-126.

Davis, J.C., 1973, Statistics and data analysis in geology, (New York: John Wiley and Sons), 313p.

Gold, C.M., 1989 (in press), Surface interpolation, spatial adjacency and G.I.S. IN: 3-D G.I.S., (J. Raper, ed.), (London: Taylor and Francis Ltd.)

Gold, C.M., 1988a, PAN Graphs: an aid to G.I.S. analysis. International Journal of Geographical Information Systems, v. 2 no. 1, pp. 29-42.

Gold, C.M., 1988b, Point and area interpolation and the digital terrain model. IN: Trends and concerns of Spatial Sciences, Second Annual International Seminar - Proceedings (June 1988) pp.133-147 + discussion (Y.C. Lee, ed.).

Gold, C.M., 1988c, Further research on Voronoi diagrams - a common basis for many applications. IN: Trends and concerns of Spatial Sciences, Third Annual International Seminar - Proceedings (Y. Bedard, ed.).

Gold, C.M., 1987, Mapping is a diffusion problem. IN: Proceedings of Carto-Quebec/Canadian Cartographic Association Meeting, Quebec, May 1987, pp. 332-362.

Gold, C.M. and S. Cormack, 1987, Spatially ordered networks and topographic reconstruction. International Journal of Geographical Information Systems, v. 1 no. 2, pp. 137-148.

Gold, C.M., T.D. Charters and J. Ramsden, 1977, Automated contour mapping using triangular element data structures and an interpolant over each triangular domain. Computer Graphics, v. 11, June 1977, pp. 170-175.

Morgan, M.A., 1967, Hardware models in geography. IN: Models in Geography, edited by R.J. Chorley and P. Haggett, (London: Methuen), pp. 727-774.

Preparata, F.P. and Shamos, M.I., 1985, Computational Geometry, (New York: Springer-Verlag), 390p.

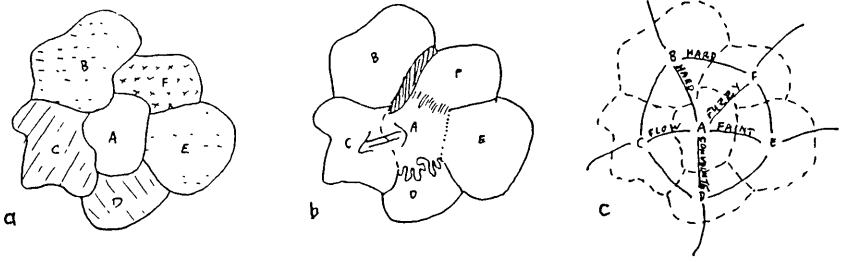


Figure 1. a) Polygon set.
 b) Possible boundary properties.
 c) Relationship triangulation (dual graph).

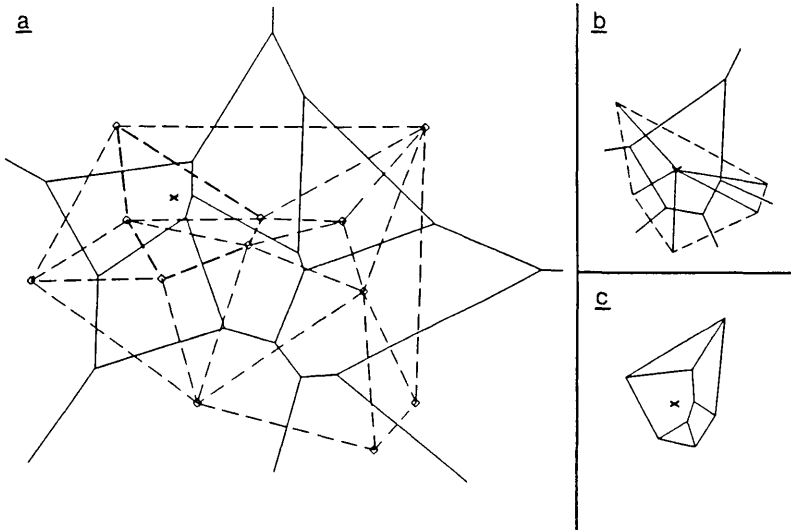


Figure 2. a) Point-Voronoi diagram and dual triangulation.
 b) Introduction of point X.
 c) Areas stolen from neighbouring regions.

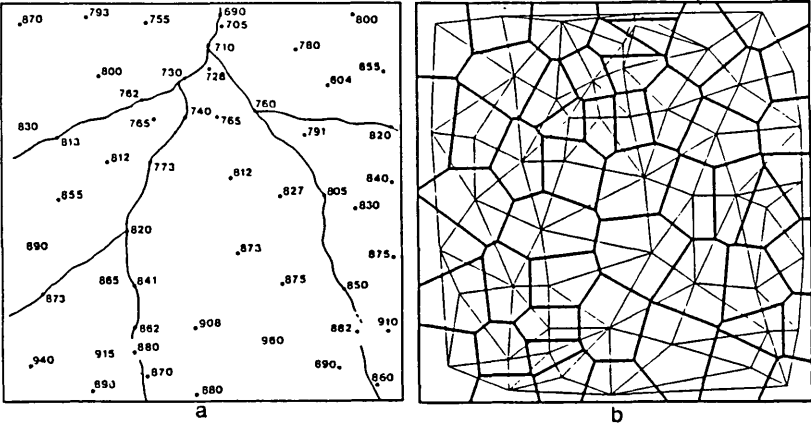


Figure 3. a) Elevation data from Davis (1973).
 b) Resulting Voronoi regions and triangulation.

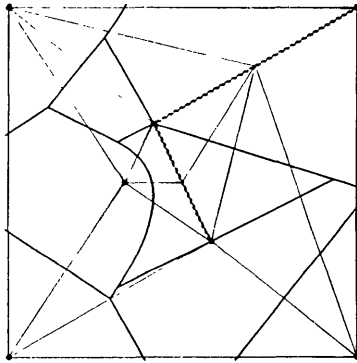


Figure 4. Voronoi regions for points and line segments.

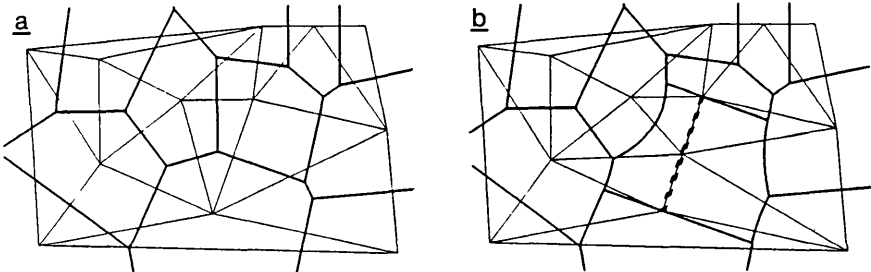


Figure 5. a) Point Voronoi regions.
 b) Insertion of a line segment.

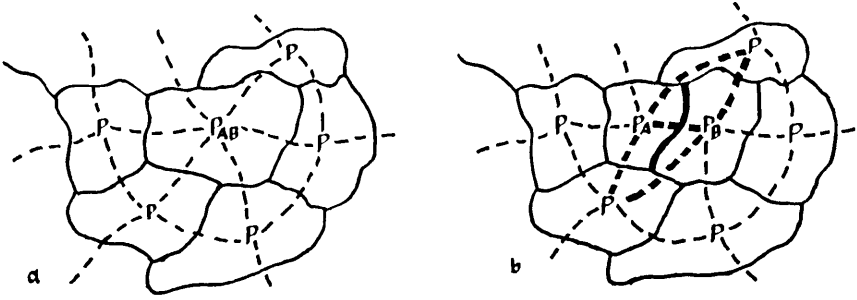


Figure 6. a) General polygon set with triangulation.
 b) Result of splitting $P(ab)$ into $P(a)$ and $P(b)$.

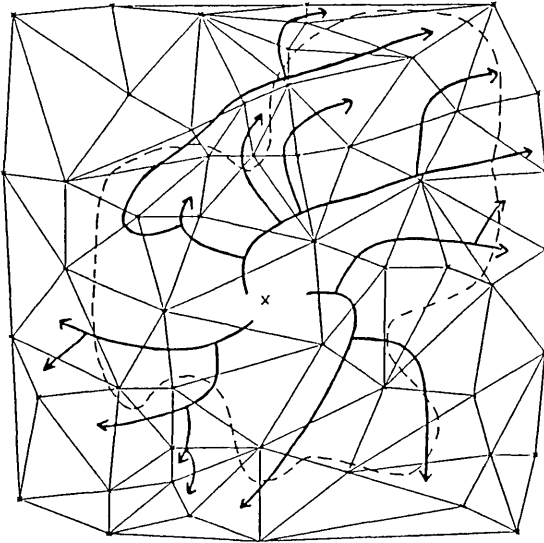


Figure 7. Triangle ordering from viewpoint X.

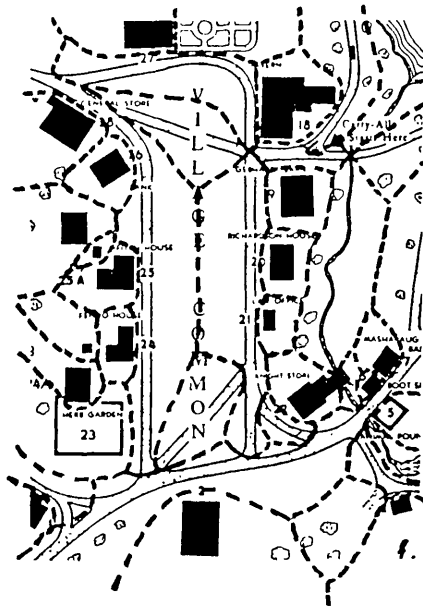


Figure 8. Map, showing map-objects and Voronoi regions.

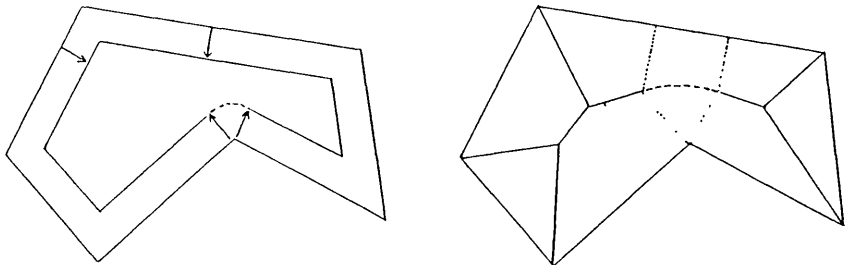


Figure 9. a) Polygon, showing wave-front propagation.
b) Internal Voronoi regions of polygon boundary.

MULTISCALE DATA MODELS FOR SPATIAL ANALYSIS,
WITH APPLICATIONS TO MULTIFRACTAL PHENOMENA

Lee De Cola

Department of Geography
University of Vermont
Burlington VT 05405
BITNET: L_DECOLA@UVMVAX

ABSTRACT

Most of the discussion in the GIS community is concerned at the highest level with the support of managerial issues and at the lowest level with engineering considerations. Scientific considerations of analytical flexibility and conceptual suitability tend to be slighted. One reason for this situation is the complex, multiscale, and "heterodimensional" world with which the scientist is concerned. The new models of spatial fractals and temporal chaos are making aggressive inroads in our understanding of complex systems, and they deserve to inform considerations of the data models that will fruitfully capture variation in space, time, and scale. I present the outline of a data model that I have found useful, along with examples of its use. The model is well suited to current advances in data organization, hardware, and parallel algorithms.

INTRODUCTION

While definitions of geographical information systems abound, little attention is given to the important differences between the managerial and scientific cultures that use such systems. Consequently, much of the GIS literature is ambiguous about whether a given system—either proposed or actual—is suitable for research as well as management. Because GIS designers and engineers, who are not necessarily the same community as scientific users, are usually concerned about efficiency and ease of use, such matters as data structure adaptability and analytical flexibility tend to be overlooked.

Substantively, the broad class of managers tend to be concerned with the support of decisions and legal issues, rather than the rules of scientific inquiry (Kaplan 1964). Managers (regional planners included) tend to work within a limited range of scales, say one or two orders of magnitude for any given task. (In what follows I shall use "scale" in its everyday, physical interpretation rather than its cartographic sense.) Moreover, at any given scale, managers focus on ideal phenomena of integral dimension: $D = 1$ points, $D = 2$ curves, and $D = 3$ areas (including most especially polygons). GIS engineers seek to serve these needs by pursuing speed, data compactness, and user friendliness in system design.

Scientists, however, and especially workers in a rapidly changing field like spatial analysis, tend to have different concerns. First their style of work tends to be flexible, tentative, often even sloppy by managerial standards: prompting the claim popular in our computer center to the effect that "If we knew what we were doing, it wouldn't be called research!" One manifestation of this flexibility is the

willingness—even the enthusiasm—of scientists to patch together disparate pieces of hardware and to use quick and dirty programs to link various software packages to achieve what they want at the moment for a particular research question.

More importantly, spatial analysts are interested in processes, particularly in many different interrelated conceptual realms: physical, biological and human. Consequently, they are trained from professional birth to relate to the world through models of process rather than the more rigid and legalistic structures of management. One consequence of this process orientation is that scientists also often want to simulate data by asking whether a suitably circumscribed set of assumptions can yield results like the complicated phenomena they find in their data. In fact, simulation can be defined as the art of getting complicated results from simple causes, but there is little evidence that GIS designers seek to support such earnest playfulness.

Another consequence of scientific interest in process is that scientists need to look at the world at a very wide range of temporal and spatial scales: often orders-of-magnitude-of-orders-of-magnitude: from millimeters to 10,000 km, from minutes to millennia. For earth scientists in particular this concern with extreme scale ranges is stimulated by a growing anxiety about the need for global monitoring, to link small causes with large effects, to test the limits of scientific ability to capture, store, analyze, and interpret vast amounts of data (IGBP 1988, p.79).

The mind of the scientist is therefore in nearly continual dialogue with his or her model, itself used to extract data from the world and to produce images, maps, tables, plots, and various statements predicting the world's complicatedness by explaining its underlying complexity. The GIS engineer, while concerned about ways of managing the world, is less preoccupied with the extent to which new systems may influence our mental images of the universe, much as did the earlier telescope or microscope (Abler 1987). While I suspect future GIS developments will overcome this limitation, for now the GIS focus is not particularly flexible, process-oriented, or multiscale.

FRACTALS AND MULTIFRACTALS

How then shall we conceive of a GIS that, while not specifically designed for scientific use, nevertheless fosters reasonable analytical ambitions? I cannot presume to fully specify such a system, first, because the above agenda is obviously broad and quite general, and second, because our collective experience with such activities is still in many ways quite limited compared with the more narrowly defined tools and questions of more traditional sciences. But if I were to set a task for a geographical analysis system it would be to address the problem of flexibly handling the input, analysis, and output of data which occur at many scales and which have fractal characteristics.

The first problem with the handling of data from a phenomenon that manifests itself over a range of scales is the requirement of large amounts of storage area and a great deal of computer power. Because

$$\text{DATA VOLUME} = \text{SCALE} \times \text{RESOLUTION} \times \text{VARIABLES} \quad (1)$$

even small increases in any of these terms will bring major jumps in amounts of data if the other is already large. Although I shall be cavalier in what follows by disregarding storage specifications, any concrete attempt to design the system I propose will demand both a lot

of hardware and a lot of engineering skill, including heavily parallel and connectionist approaches.

The second problem with scale is that the mind manifests a certain scale inertia: we tend not to make ready mental shifts in magnitude. Although the forest/tree distinction is familiar to everyone, few people recognize that for some analytical purposes it helps to conceive of forests and trees as part of a conceptually unitary phenomenon in which many processes interact at many scales to reveal structures that are, not at all paradoxically, "scaling," i.e. ranging from the nearly infinitely large to the nearly infinitely small (and quite intense). So we talk about trees and forests (read neighborhoods and urban systems, rocks and mountains) as though they were different entities rather than mental images of the same thing. The fractal paradigm is helpful here.

Fractals are phenomena which are self-similar: images and measurements of fractals taken at one scale tend to be similar to images and measurements at another scale. The problems this presents for traditional science can be understood by considering the following history. The goal of classical science has traditionally been to look for regularities (linearities or log-linearities) in data: this often entails measuring variables operationally defined within a narrow range of scales, then modelling relationships, and finally sweeping what is left into an error term. Fractal research challenged this approach by arguing that in many realms virtually everything may be "error". Early geometrical research demonstrated how ideal fractals were scale-invariant, implying that the new regularities were captured by D , a parameter which could be used to "explain" the phenomenon. This conclusion, while naive, is less egregious than the notion that scale is unimportant. Scale (like money) matters, and must be explicitly part of any spatial analysis system.

Later work on stochastic fractal phenomena generated images and data that were more realistic, to be sure, but also spawned a number of different fractal dimensions depending upon the model or the aspect of the phenomenon under study (Stanley 1986). This tends to be upsetting unless one realizes that different facets of a process (say the perimeter of a region versus its "mass") will have different fractal dimensions. The lesson here is for a spatial analysis system to allow clarity about these facets and their measures.

The latest phase of research—and we are now at the cutting edge because an aggressive game is being played with the real world—has begun to focus on real, multifractal phenomena whose fractal dimensions vary with time, with space, and (although it may seem paradoxical) with scale itself (Feder 1988). Specifically because the universe is made of systems (molecules, people, planets) with characteristic lengths, system behavior changes with scale. Much of this work is still highly theoretical, but some is empirical: perhaps the most relevant to spatial analysts is the research of Lovejoy and others on turbulence in meteorological systems (Gabriel et al 1988) as well as that on earthquakes (Kagan 1980).

The key notion here is that seismological, meteorological, and, to be boldly hypothetical, cultural systems are intermittent processes generating structures in a dissipative cascades from larger scales to smaller. At the largest scale such systems generate space-filling structures of $D \sim 3$, while at the smallest we find "singularities" of $D \sim 0$. The lesson for geographers, in particular, is that such key

descriptions as area, intensity, variability, complicatedness, etc. will vary not only with time and space but also with scale. The challenge remains to extract regularity from these data by being able to exert control, in the quasi-experimental sense of collecting lots of data, over scale.

Geographic science is a full participant in this revolution. Geographers have made early and enthusiastic use of these notions and have revealed the multifractality of terrain over space (Mark and Aronson 1984, Roy et al 1986), of coastlines over scale (Goodchild 1980), of sedimentation over time (Plotnick 1986), of variation over scale (Woodcock and Strahler 1987) and of point pattern over density (Harvey 1968, De Cola 1987).

The positive side of all this is that quite parsimonious fractal models are yielding important results in a wide variety of fields and that various dimensions, provided they are operationally defined and displayed for a range of times, places, and scales, are extremely powerful descriptors of real phenomena. Moreover, fractal theory teaches that we often need fewer variables than we thought, which mitigates the Devil's bargain reflected in Equation 1. Still, when it comes to memory and speed, more is quite clearly more, and can sometimes compensate for lack of conceptual rigor.

DATA MODELS

Any geographical analysis system for scientific research must therefore be capable of gathering, managing, and displaying data from fractal phenomena. Rather than narrowly specifying a data structure for such analysis, however, my ambition is more modest. Peuquet (1984, p. 69) defines a data model as "a human conceptualization of reality, without consideration of hardware and other implementation conventions or restrictions." Certainly the conceptual specifications that follow, along with the examples, can be translated into the syntax of any language (such as C or Pascal) permitting variant and dynamic records as well as recursion. Greater specificity would obscure a heuristic approach to structure and an algorithmic approach to process (Smith 1987).

The above outline of fractal research calls for a discussion of four issues. First, we should be clear about the topological realm in which we are working. To begin, I shall not be concerned with phenomena in a third (physically vertical) dimension, although it is clear that the analysis of terrain data is basically a question of analyzing variation in space (Weibel 1988). In terms of integral dimensions, the topologies of the phenomena to be examined range from simple points to space-filling areas, but note how differently these topologies are treated:

<u>DIMENSION</u>	<u>GEOMETRY</u>	<u>SPATIAL ANALYSIS</u>	<u>REMOTE SENSING</u>	<u>GIS</u>
0	Point	Event	Location	Location
1	Curve	Link or Perimeter	Pixel edge	Vector
2	Area	Region	Pixel	Polygon

I shall not attempt to do more than acknowledge this incommensurability except to note that what sometimes appears to be a culture conflict may be quite profound: the data structures of GIS are based on atomic units

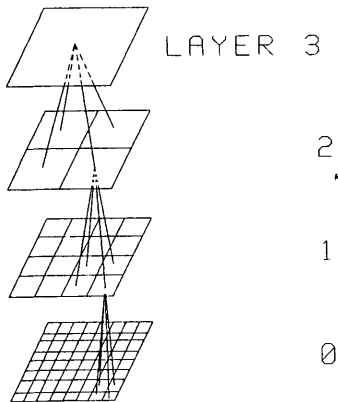
that are not only too small to be detected with remote sensing techniques, they simply do not exist in the two dimensional system ($D = 2$) used by sensors. This is not to make the trivial observation that a benchmark location, a road centerline, or a state boundary is invisible; but rather that they exist in a space of dimension too low even to be incorporated in such a system. Now remote sensing projects 3- and 4-dimensional phenomena into 2-dimensional data, but unfortunately projection only works one way. It is impossible to "project" 1-dimensional GIS entities into a 2-dimensional map. (For a related discussion of this problem see Lovejoy's (1985) discussion of meteorological networks.)

A second area requiring specificity is the distinction between scale and resolution. Although the term scale is used cartographically to represent the size that the representation of an object on a map bears to its "real" size, I choose here to use scale in its more commonplace (and physics) sense of the characteristic size of the real object. In contradistinction, I shall use resolution to mean the smallest areal unit at which a distinction of spatial variation can be made. On the one hand, a data structure must be able to incorporate measurements at the highest resolution made available by sensors (consequently it must be big) while at the same time capturing or permitting the creation of structures of often enormously large scale (Equation 1 above). The data model must also be able to integrate measurements from sources with different resolutions. On the other hand, this need for breadth of scale is mitigated by the fact that scientific data need not be stored with great absolute locational accuracy (Burrough 1988). Third, the data model must reflect clarity of thinking as well as flexibility about such things as features, objects, and entities. I have opted below to call homogeneous clusters of cells "regions," but readers are unlikely to agree that the world can be partitioned into such sets. While the regions are constructed "bottom-up" by aggregating cells, how they are generically defined, functionally specified, or identified by name will be a function of their scale as well as that of the analysis.

Finally, the system also needs to operate "top-down," beginning with a dataset and subdividing it into subsets of greater homogeneity or requiring small-scale analysis. Quadrees and their variants are a popular approach to this problem, and I would argue that the full pyramid represented by a complete quadtree structure suggests the approach called for. It should also be recognized that the size of a quadtree is a direct reflection of fractal dimension (Samet 1984, p. 227).

EXAMPLES

In the present case, we begin with grid dataset, conceived of as a lattice of $(2^L)^2$ 0-level cells $x_{0,i}$, where $i = 1, \dots, (2^L)^2$. Let there be a value $f(x_{0,i})$ assumed to be univariate and to be an explicit, monotone function of some underlying measurement: i.e. f could be an observation of events people, photons, trees, votes, etc. Let a λ -level lattice or layer of $(2^{L-\lambda})^2$ cells be constructed with $f(x_{\lambda,i}) = \sum f(x_{\lambda-1,j})$, where the summation is over the $j = 1, \dots, 4$ children cells of $x_{\lambda,i}$. See the following figure:



Three things should be noted. First, for layer $\lambda+1$ to be a true aggregation of layer λ it is important that f be a linear function of the measured phenomenon, otherwise we need a way to recover the measurement from the data (Richards 1986). Second, the possibility of f being multivariate is a complexity I shall not explore. Third, although the present lattice is of dimension 2, the lattice could be of side $(2^L)^3$ or even dimensionally larger, to include time, etc. A histogram of $\{f_\lambda\}$ would describe each layer λ , and if f were a digital transformation of the underlying count then the abscissa of this histogram would be limited (as in the case of a Landsat band to 2^8 values). In any case various parametric (moments) and nonparametric statistics will also be used to characterize the layers. These statistics would indicate the presence of cells with unusual concentrations (or absences) of events. Higher, more aggregate, levels could be used to scan the image for intense activity. Next, let $t \in [0, \max\{f_0\}]$ be a threshold. The variable t could also stand, in increasing order of complexity, for an interval, a subset, or even the intersection of multivariate subsets, as well as labelled classes. Obviously this rapidly complicates things but does address the task of image classification and labelling (Campbell 1987). At any level λ consider the regionalization $F_{\lambda,t} = \{x_{\lambda,i} : f(x_{\lambda,i}) \geq 2^\lambda t\}$, i.e. the set of all super-threshold cells (De Cola 1989).

This image segmentation creates a list of disjoint and unconnected regions $\{E_{\lambda,t,k} : k = 1, \dots, n(\lambda,t)\}$, where $n(\lambda,t)$ is the number of such regions. Each region E can be characterized at least by its location, size (number of cells), and perimeter. We may store this list either in its entirety or, by the use of dynamic variables, in bins containing the above descriptive information. The number, sizes and perimeters of these regions can be used to compute the fractal dimension $D(\lambda,t)$ as well as the Pareto scale parameter $a(\lambda,t)$ for the layer λ , both of which can be expected to be a declining function of t (Lovejoy and Schertzer 1988). From the point of view of memory considerations, it should be noted that in general $\partial^2 n(\lambda,t) / \partial t^2 < 0$, i.e. the number of regions tends to a maximum for some midrange value of t (roughly that value of t for which $p(x_\lambda \in F_{\lambda,t}) = 1/4$ (De Cola 1989)). This bottom-up approach yields regions which are explicitly a function of threshold, of layer, and of such sensing characteristics as resolution, so that we may examine the extent to which regional description and appearance reflects these characteristics. Note therefore that we may explicitly examine resolution effects both as artifact and as explanatory variable.

An example of the multi-scale imaging proposed here is shown in Figure 1, which displays results of a random walk simulation of 1000 steps on a 32^2 torus. Shown in 1d) are all cells visited at least $t = 2$ times, in 1c) all 2×2 cells visited at least $2^2 \cdot 2 = 8$ times, etc. For this experiment $D(t=2) = 1.41$. It is helpful to think of each of these figures as an image in itself and not a "defocusing", etc. of some "better" resolution data. Sometimes we are interested in forests, sometimes in trees, still other times in leaves. Each layer tells us something about the process at that scale, characterized by fractal and size distribution parameters, and each threshold generates different statistical characterizations.

Another example of this approach, this time from empirical research, is shown in Figure 2. Figure 2a) presents all of the URBAN-classified $(31.81 \text{ m})^2$ pixels from a from a 2048^2 -pixel Landsat image of Northwest Vermont (De Cola in review). Figure 2b) (at the scale of 2a)) locates all level-0 regions of size ≥ 15 pixels. But another way of looking at this information is shown in 2c), all level-4 regions; i.e. all clusters of $2^4 = 16$ -sided cells in the study region. (Note that $t = \text{URBAN}$ is not a threshold but an imputed land use; nominal values require a form of aggregation different from simple summing (De Cola 1989). This aggregation process reduced the number of regions from the 39,000 of the level-0 image to a manageable 325 in the level-4 image. These regions were then used to estimate the populations of "towns" in Franklin County VT, with results that were more reliable than those derived from the use of level-0 single-pixel regions (De Cola 1988 and see Tobler 1969).

So much for a "bottom-up" approach; next we turn to the recursive subdivision of the highest cell at level L . Each layer $L-\lambda$ consists of cells representing locations x which can be described in terms of such parameters as $D(\lambda, t)_x$, where the subscript implies locational specificity within the layer. This technique applies as well to Pareto regional size distribution statistics, to the size of the largest region in cell x , etc. Although these parameters can be presented in tables and plots, perhaps the most interesting way of displaying them is as maps. Figure 2d), for example, is a map of the fractal dimension of URBAN pixels from the Vermont study. The scene has been divided into $(2^3)^2 = 64$ cells, bringing the analysis down to the $\lambda = 11 - 3 = 8$ level. The fractal dimension $D(8, \text{URBAN})_x$ is represented by the height of each point. Lack of variation in D_x over space would be a necessary indicator of texture.

The data model presented here is obviously extremely flexible, allowing the scientist 1) to move among scales from layer to layer, 2) to explore the effects of varying threshold, and 3) to roam spatially within layers from cell to cell. As such, it affords ready access to parameters for the scale and location examined, as well as access to images at lower levels and maps at higher. The key is obviously the cell, which is not only the constituent of a region built up from below but also the location of spatial information for a window of subdivided space. For example, in the example above, the URBAN spectral classification was determined at level 0, while the identification of actual "urban" features was made at level 4, and the exploration of spatial variation in URBAN fractal dimension was made at level 8. The difference between the urban names here is intended explicitly to recognize that URBAN is a pixel group operationally defined by the specification of a spectral classification process, while "urban" is a word I have chosen to denote level 4 regions with a specific spatial morphology (connectedness, size, disjointedness, and form).

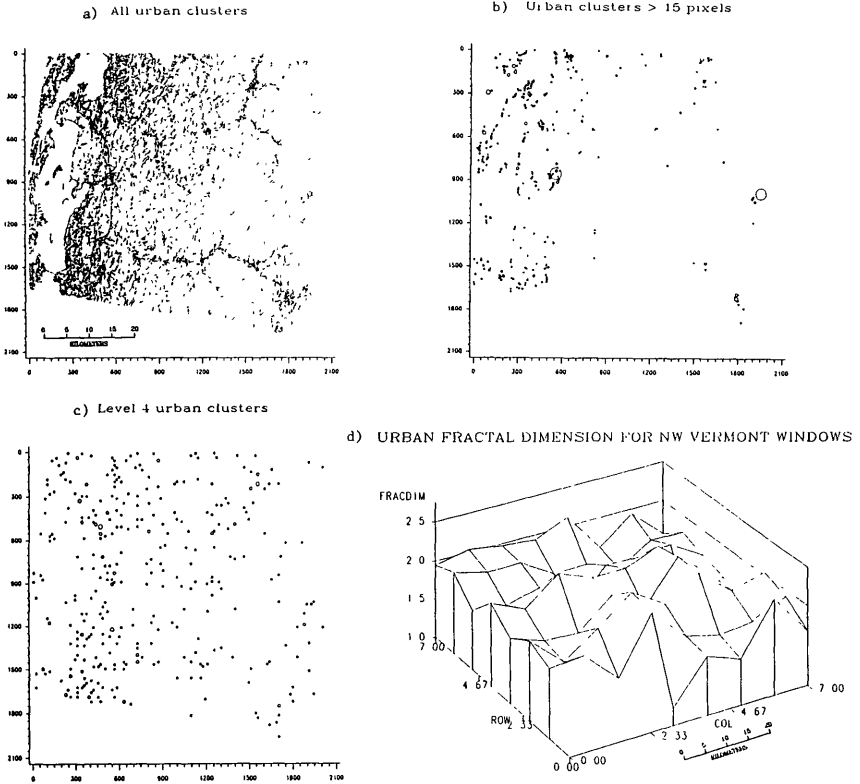


Figure 2. Images and a map of URBAN pixels from a Landsat scene for Northwest Vermont.

PROSPECT

Admittedly, it is no easy feat to translate the above requirements into a full-blown geographical analysis system for the study of events, interactions, and regions in space and time, but we are moving in this direction. At least three kinds of developments are cause for optimism. First, data structures, such as quadtrees and iterated function systems (Barnesley 1988), as well as data hardware, such as faster and more capacious chips as well CD-ROM (Lambert and Ropiequet 1986) give us powerful command over and broad access to large numbers of measurements. Second, sheer improvements in computational speed are always welcome, but probably the greatest advances will come from parallel and connectionist architectures and algorithms (Toffoli 1987 and Mower 1988). There seems no reason why a parallelist approach to spatial data (Bhaskar, Rosenfeld and Wu 1988) cannot be adopted to the multidimensional analysis of digitized spectral data as well. Finally, multifractal approaches to real world phenomena offer new ways of integrating multisource data in ways that make analysts less burdened by hitherto supposedly incompatible resolutions. While I am not optimistic about the near-term integration of vector ($D = 1$) and raster ($D = 2$) data, I am excited about the fact that scientific research can only flourish in these tumultuous times.

REFERENCES

- Abler, R.F. 1987. "What shall we say? To whom shall we speak?", Annals of the AAG, 77(7):511-524.
- Barnesley, M.F. 1988. In Peitgen, H-O and Dietmar Saupe 1988. The science of fractal images. New York: Springer-Verlag.
- Bhaskar, S.K., A. Rosenfeld and A.Y. Wu. 1988. "Parallel processing of regions represented by linear quadtrees." Computer Vision, Graphics, and Image Processing, 42:371-380.
- Burrough, P.A., W. van Deursen and G. Heuvelink 1988. "Linking spatial process models and GIS: a marriage of convenience or a blossoming partnership?" GIS/LIS'88, San Antonio, pp. 598-607.
- Campbell, J.B. 1987. Introduction to remote sensing. New York: Guilford.
- De Cola, L. 1987. "Fractal analysis of digital landscapes simulated by two spatial processes." Paper presented at AAG Annual Meeting, Portland OR.
- _____ 1988. "Fractal estimates of place populations." Paper presented at AAG Annual Meeting, Phoenix, AZ.
- _____ 1989. "Pareto and fractal description of regions from a binomial lattice" Geographical Analysis.
- _____ in review. "Fractal analysis of a classified Landsat scene."
- Feder, J. 1988. Fractals, New York: Springer-Verlag.
- Gabriel, P., S. Lovejoy, D. Schertzer and G.L. Austin 1988. "Multifractal analysis of resolution dependence in satellite imagery." Geophysical Research Letters, 15(2):1373-1376.
- Goodchild, M.F. 1980. "Fractals and the accuracy of geographical measures." Mathematical Geology, 12(2):85-98.
- Harvey, D.W. 1968. "Pattern, process and the scale problem in geographical research" Transactions of the Institute of British Geographers, 45:71-78.
- International Geosphere-Biosphere Programme 1988. Global change: a plan for action. Stockholm: IGBP.

- Kagan, Y.Y. and L.Knopoff 1980. "Spatial distribution of earthquakes: the two-point correlation function." Geophysics Journal of Royal Astronomical Society, 62:303-320.
- Kaplan, A. 1964. The conduct of inquiry. San Francisco: Chandler.
- Lambert, S. and S. Ropiequet 1986. CD-ROM: the new papyrus Redmond WA: Microsoft.
- Lovejoy, S. and D. Schertzer 1988. "Extreme variability, scaling fractals in remote sensing: analysis and simulation." in Muller, J.C. (1988), pp. 177-212.
- Lovejoy, S., D. Schertzer and P. Ladoy 1985. "Fractal characterization in homogeneous geophysical measuring networks." Nature, 319(6048):43.
- Mark, D.M., and P.B. Aronson 1984. "Scale dependent fractal dimenions of topographic surfaces: an empirical investigation with applications to geomorphology and computer maping" Mathematical Geology, 16(7):671-683.
- Mower, J.E. 1988. "A neural newtork approach to feature recognition along cartographic lines." GIS/LIS'88, San Antonio, 1:250-255.
- Peuquet, D. 1984. "A conceptual framework and comparison of spatial data models." Cartographica, 21(4):66-113.
- Plotnick, R.E. 1986. "A fractal model for the distribution of stratigraphic hiatuses." Journal of Geology, 94(6):885-890.
- Richards, J.A. 1986. Remote sensing digital image analysis. New York: Springer-Verlag.
- Roy, A.G., Ginette Gravel and Cline Gauthier 1986. "Measuring the dimension of surfaces: a review and appraisal of different methods." Autocarto, 8:68-77.
- Samet, H. 1984. "The quadtree and related hierarchical data structures." ACM Computing Surveys, 16(2):187-260.
- Smith, T.R., S. Menon, J.L. Star, and J.E. Estes 1987. "Requirements and principles for the implementation and construction of large scale geographic information systems." International Journal of GIS, 1(1):13-32.
- Stanley, H.E. and N. Ostrowsky 1986. On Growth and Form: Fractal and non-fractal patterns in physics, Boston: Martinus Nijhoff.
- Tobler, W.R. 1969. "Satellite confirmation of settlement size coefficients." Area, 1(3):31-34.
- Toffoli, T. 1987. "Pattern recognition and tracking by texture-locked loops." MIT Lab for computer science.
- Weibel, R. 1988. "Automated terrain classification for GIS modelling." GIS/LIS'88, San Antonio, pp. 618-627.
- Woodcock, C.E. and A.H. Strahler 1987. "The factor of scale in remote sensing." Remote Sensing of Environment, 21(3):311-332.

THREE-DIMENSIONAL GIS FOR THE EARTH SCIENCES

Dennis R. Smith & Arthur R. Paradis
Dynamic Graphics, Inc.
2855 Telegraph Ave, Suite 405
Berkeley, CA 94705

ABSTRACT

Earth scientists are frequently confronted with problems that involve 3-dimensional phenomena, but up until now the main computerized geoprocessing tool available to them has been the 2-dimensional GIS. New 3-dimensional geoprocessing capabilities are being developed and will be in their hands in 1989. These systems address a class of problems that could not be dealt with before and will provide answers to questions that we did not realize we could ask.

INTRODUCTION

Different application groups have been using Geographic Information Systems (GIS) for various reasons. In talking to these people you soon realize that GIS means different things to different people. Over the years there have been attempts to define what a GIS is and how it is used. Recent GIS reviews include Cowen (1988) and Clarke (1986). For the purposes of this paper we will use the following definition of a GIS. A GIS is a software system that contains functions to perform input, storage, editing, manipulation, analysis and display of geographically located data.

Up to now the main uses of a GIS have dealt with data on the earth's surface. If the data was above or below the surface it was conveniently projected to the surface. This allowed the system to deal with everything in a 2-dimensional format. Early GIS's often used a data structure of regular grid cells but current systems seem to favor polygons. All of these deal with many flat files that are oriented over the same location of the earth. Sometimes a system could draw a perspective view of the surface and even present data on the surface, below the surface and above the surface. These presentation techniques still deal with flat files but add the capability to present the data in what we will call a 2 1/2 -dimensional format.

WHY 3-DIMENSIONAL GEOPROCESSING

Many earth scientists who have tried their hand at geoprocessing have come up short. The use of flat, 2-dimensional files does not fit their needs. These scientists are usually dealing with geology, geophysics, meteorology, hydrology, mining, ground water, hazardous contaminations, and the like. These phenomena are 3-dimensional in nature and when you try to fit them into 2-dimensional systems you can not accurately model, analyze or display the information.

To help explain things throughout this paper we will use an example from a situation that we all hear about these days; the problems with hazardous chemicals in the ground. At this particular site they discovered, in the ground, PCB concentrations that were above safe levels. This indicated to the site owner that an expensive undertaking was necessary to first determine the extent of the problem and then to correct the situation.

It is impossible to model, analyze or display this situation, with any satisfaction, when you are using a 2-dimensional tool. You might be partially successful in using stacked 2-dimensional data layers but you are basically forced to ignore the fact that the phenomena is actually 3-dimensional. Applying a 2-dimensional tool to 3-dimensional situations limits the scientist's work in many ways. It is not possible to accurately model the vertical relationships between the stacked 2-dimensional layers. It is not possible to perform true 3-dimensional analytic operations between different models. It is not possible to accurately visualize the 3-dimensional situation and make decisions about the data.

THREE-DIMENSIONAL DATA

Let's go to our example site and take a look at the source data that is available. Contamination was discovered in the ground and new wells were drilled to gather additional data. Samples were taken at various locations down these wells and sent to the lab for analysis. High levels of PCB were discovered. The site was contaminated and had to be cleaned up. The PCB values were reported in a tabular fashion with the geographic location of each sample. With a 3-dimensional data set you need to know X,Y,Z&V where X,Y,Z give the location of the property, and V is the value of the property at that location. The property in this situation is the concentration level of PCBs. A portion of the data file is shown in Table 1.

Well-ID	X-coord	Y-coord	Elevation	PCB-level
2002	-1165	763	-80	0.33
2002	-1165	763	-140	0.16
2002	-1165	763	-200	0.66
2003	-1140	743	-20	0.05
2003	-1140	743	-80	0.06
2003	-1140	743	-140	0.09
2003	-1140	743	-200	0.13
2004	-1165	718	-20	0.13
2004	-1165	718	-80	0.45
2004	-1165	718	-140	0.10
2005	-1200	743	-20	0.13
2005	-1200	743	-80	0.72
2005	-1200	743	-140	0.09
2005	-1200	743	-200	0.33
2006	-1175	600	-20	0.19
2006	-1175	600	-80	0.22
2006	-1175	600	-140	0.14

Table 1. Portion of the Source File

With a 2-dimensional system the X&Y location of each well can be displayed and selected horizontal planes can be utilized in an attempt to model, analyze and display slices through the earth. With a 3-dimensional system the data is input, edited, modeled, analyzed and displayed in its true 3-dimensional form. Figure 1 shows a display of source data with a 3-dimensional system.

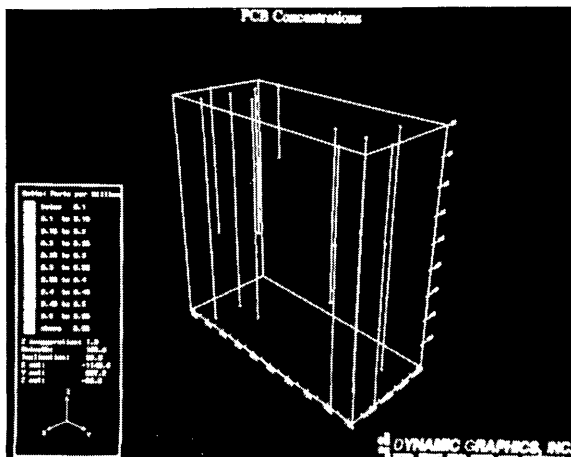


Figure 1. Three-Dimensional Source Data

THREE-DIMENSIONAL MODELING

In 2-dimensional systems the user often models the scattered or randomly located source data into a uniform or regular data structure. These have typically been uniform grids or triangles. The reason for this modeling is that the resources required to analyze and display data that is located in a scattered format is significantly higher than dealing with data in a regular format. Each time the scattered data needs to be contoured, or each time volumes need to be calculated, a modeling step would have to take place. The intention is to perform the modeling step once. As long as the mathematical model fits the physical model, the savings in resources are worth it.

The situation with 3-dimensional phenomena is similar. It is more efficient to model the scattered data once onto a uniform grid than to deal with it in its scattered format. The objective of the modeling step is to apply a mathematical model that best fits the physical model. The model will never be truth. In situations with subsurface problems there is usually only a limited amount of data available because of the high cost of drilling wells to collect new data. The analysts never have as much data as they would like. It's not like topography, where you can go out and stand on the site and see it first hand.

Many phenomena in nature follow a model known as minimum-tension. A computer-generated minimum-tension model can be calculated using an iterative tension reduction method. See Briggs (1974). If there is no other information known about the phenomenon except for its value at particular spot locations, then the minimum-tension algorithm provides a smooth, unbiased model of the data. If any additional facts are known about the phenomenon then, of course, that has to be taken into account by applying another model which better fits the situation.

For example, if the phenomenon is moving through a ground water zone and it is known that the zone has an East-to-West flow, then an appropriate flow model should be applied, not a minimum-tension model. Sometimes a flow model will create a non-uniform grid and a minimum-tension algorithm can then be used to model the non-uniform data onto a uniform-grid. This is done when the data needs to be correlated with other non-uniform data sets, or when particular display techniques need to be used that require a uniform grid.

Another modeling technique involves the use of geostatistics to provide the scientist or analyst with information. Geostatistical models, such as kriging, are often used for applications in mining

or petroleum exploration. Geostatistical routines are not used to develop a mathematical model of a physical phenomenon like the minimum-tension model does, but rather they strive to develop a block average description of the phenomenon. As the blocks are larger the results seem more valid.

DATA EDITING

The ability to input source data and run a model is very useful, but often the user is confronted with the need to edit the data. Sometimes the source data needs to be queried and/or edited, and other times the model results need to be queried and/or edited. Editing tools, often involving interactive graphic editors, can be applied to 2-dimensional data without a great amount of difficulty, but with 3-dimensional data the problem is much more difficult. Working in 3-dimensions the tools have to be more helpful to the user and have to be graphically more powerful. It is not easy to point and query data locations in 3-dimensions and it is more difficult to edit the source data and then remodel around it.

THREE-DIMENSIONAL ANALYSIS

One of the more simple techniques that can be used to analyze 3-dimensional data is to apply a set of grid operations. These operations could include such things as:

- grid-to-grid mathematical calculations
- grid refinement
- grid smoothing
- back interpolation
- trend grids

These grid operations would provide a user with a basic set of tools to perform a wide range of analytic functions. Three-dimensional grid models of permeability, porosity, temperature and pressure could be compared, correlated and analyzed together to determine the most likely locations for oil to be found. Hazardous chemical plumes in the ground could be analyzed over time to determine the movement of the plume and any changes in its size or chemical make-up.

When we first think of 3-dimensional problems we often imagine applying analytic tools similar to those we are familiar with in 2-dimensions. This is a very reasonable assumption to start with. However, a 3-dimensional gridded model of a particular phenomenon does not always provide us with the data structure

that we need in order to perform some of these operations. One of the solutions to this is to develop 3-dimensional iso-surfaces through the 3-dimensional grids similar to the way we locate 2-dimensional iso-lines (contours) through 2-dimensional grids. An example of an iso-surface is shown in Figure 2.

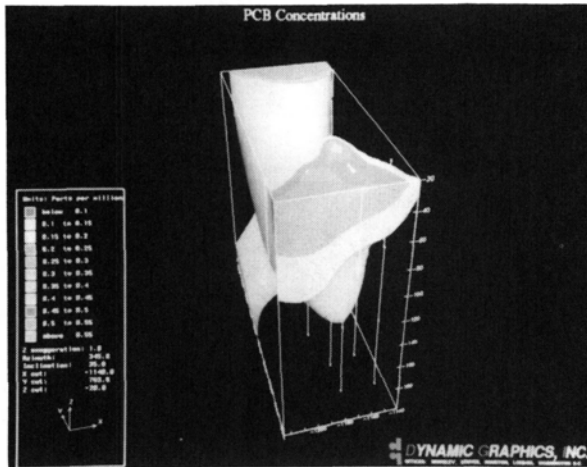


Figure 2. Example Of An Iso-surface

The iso-surface is a polygonized data structure which is positioned through the 3-dimensional grid where the level of the phenomenon is of equal value. An iso-surface in 3-dimensions is similar to an iso-line or contour line in 2-dimensions. The iso-surface is given its shape by forming small triangular polygons through the gridded data and then connecting these triangles together to form a 3-dimensional surface of equal value, or an iso-surface.

Having the phenomenon defined by user-selected iso-surfaces provides the scientist with an additional set of analytic capabilities. Accurate volumes can now be calculated. Iso-surfaces can be intersected by performing 3-dimensional polygon intersection. This operation could allow a scientist to accurately model the movement of a contaminated plume through the ground.

Iso-surfaces can be constrained or limited above and below by 2-dimensional surfaces. This could be used to generate an accurate geologic model where particular materials are limited by geologic structures and faults. Polygons defining features on the

earth's surface, such as lease tract boundaries, could be used to cut down through the 3-dimensional iso-surfaces by using a 3-dimensional polygon intersection routine. This would be useful, for example, when a user wants to calculate volumes under lease tracts, or when some underground phenomenon needs to be associated with particular land-use features.

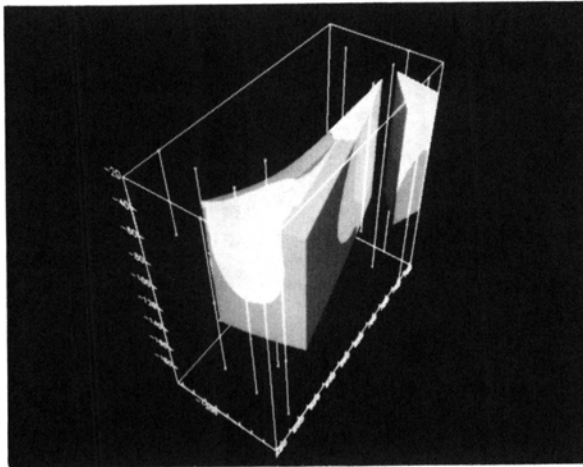


Figure 3. Iso-surfaces Cut By Polygons

THREE-DIMENSIONAL DYNAMIC DISPLAYS

One of the more powerful and useful functions of a 3-dimensional geoprocessing system is its ability to display information in ways that have never before been seen. This provides the user with a scientific visualization tool that allows him to better understand the phenomenon he is studying and to make more informed decisions about it. The display capability needs to include dynamic movement of the graphic and dynamic selection of options. There is a tremendous advantage in having dynamic capabilities in a 3-dimensional system because of the complexity of the problems. Many of the relationships and features of 3-dimensional phenomena simply cannot be comprehended by the users without these tools.

One of the basic elements of any geographic display is the need to accurately identify the geo-referencing system. The user needs to know where things are, not only in X and Y coordinates, but also in Z coordinates. The user can easily get lost when the system provides the capabilities to rotate the model left, right, up or down. The system needs to provide the user with an appropriate geographic referencing system.

Users need to be able to see and browse thru displays of the scattered source point data. They also need to be able to examine and understand the 3-dimensional grid values and how they relate to the source data. Another very informative display of the 3-dimensional model is a colored cube display which presents ranges of the values in the model.

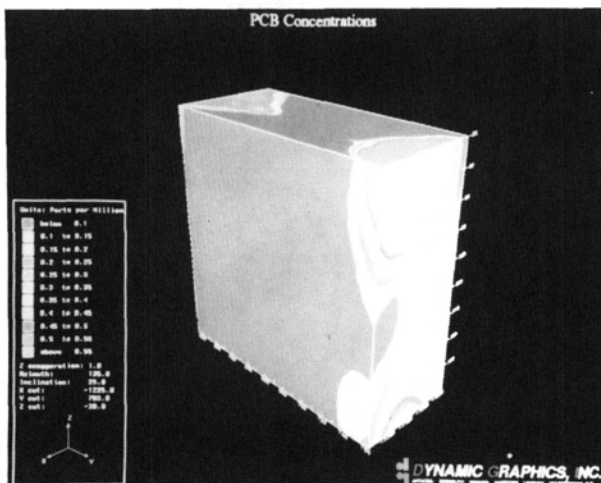


Figure 4. Full Cube Display

The user needs to be able to slice off edges of the cube display to get views of the inside of the model in order to better understand what the model really looks like.

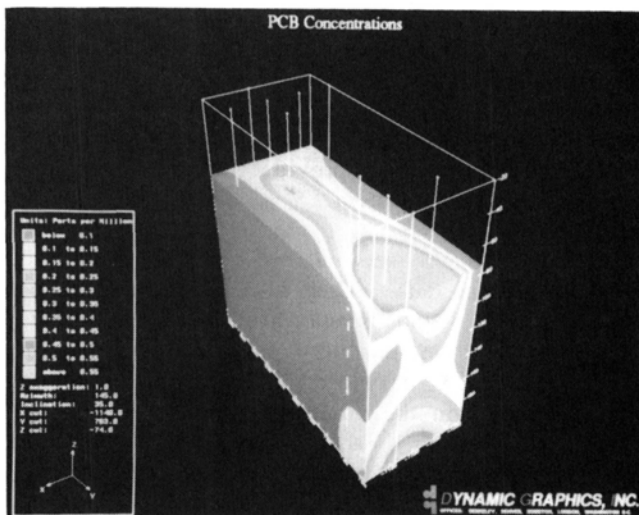


Figure 5. Sliced Cube Display

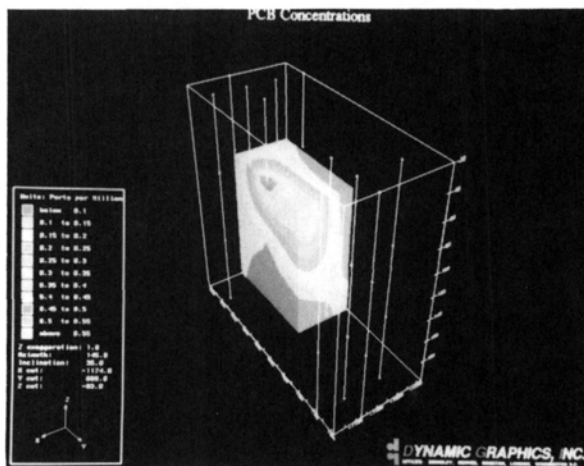


Figure 6. Sliced Cube Display

Often the user is searching for a particular value in the data and the system needs to provide tools to locate and display this value.

For example, when the ground is contaminated with PCB's as in the sample used here, the user is trying to determine if the values found are above the safe levels for that particular chemical, and if so, what the volume is and where it is located. In a 3-dimensional geoprocessing system the user could select a particular iso-surface level, have the display generated, and then dynamically slice through it to gain a full understanding of the situation.

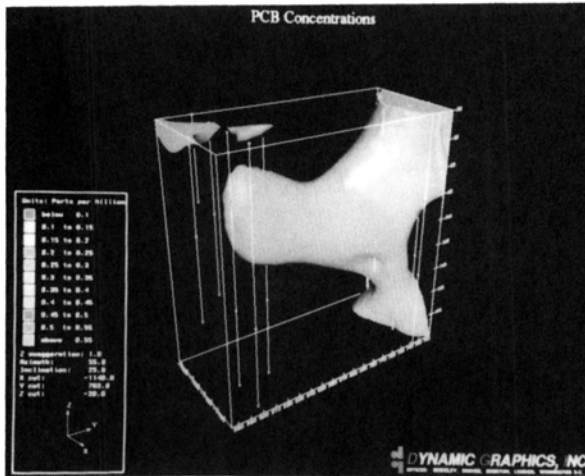


Figure 7. Iso-surface Display

Phenomena in 3-dimensions are difficult to understand and all of the display functions in a 3-dimensional geoprocessing system need to work together to provide the users with the maximum utility. The user needs to be able to see the source data, to select different iso-surface levels, to assign colors to these levels, to slice edges from the model, to peel off iso-surfaces, to rotate around the display and to zoom in and out.

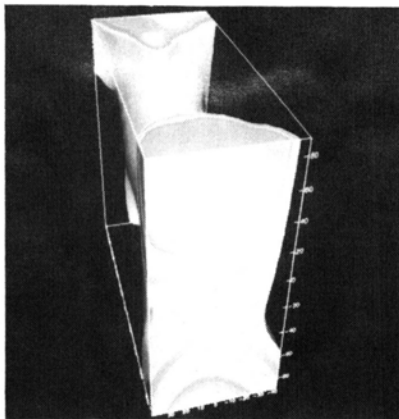


Figure 8. Concentrations Above a Value

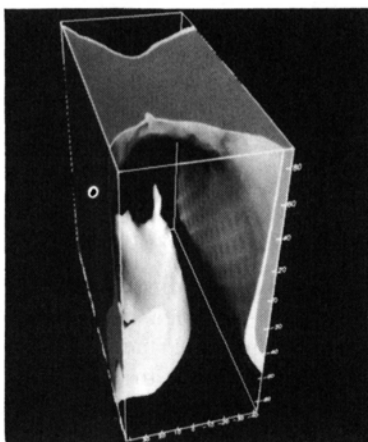


Figure 9. Concentrations Below a Value

HARDWARE ISSUES

The capabilities needed to work with 3-dimensional data are greatly enhanced by the hardware functions in new graphic workstations. Three-dimensional modeling is compute intensive and is well suited for the new high-performance 3-D graphic workstations. The scientific visualization aspects of a 3-dimensional geoprocessing system are only available by utilizing the 3-dimensional graphic functions in the new workstations. No one view can properly communicate to the user what is going on in a 3-dimensional model. Fortunately for the user community, 3-D workstations are becoming more common in the work place and prices are dropping.

CONCLUSION

These new 3-dimensional geoprocessing capabilities are addressing a class of problems that could not be dealt with before. Manual methods can be successful with 2-dimensional problems and many of these are now addressed by computerized systems. The 3-dimensional problems in the earth sciences are generally too complicated to do by hand and computerized systems to date have not been that successful. Applying 2-dimensional tools to 3-dimensional problems has been only moderately successful at best. As the new 3-dimensional geoprocessing tools get into the hands of the users, answers will be discovered to questions that we currently don't understand or even realize we can ask. For earth scientists the move from 2-dimensional geoprocessing into 3-dimensional geoprocessing will be both exciting and rewarding.

REFERENCES

- Briggs, I.C., 1974. Machine Contouring Using Minimum Curvature, Geophysics, Vol 39 - No 1, pp 39-48.
- Clarke, K.C., 1986. Advances in Geographic Information Systems, Computers, Environment and Urban Systems, Vol 10, pp 175-184.
- Cowen, D.J., 1988. GIS versus CAD versus DBMS: What Are the Differences?, Photogrametric Engineering & Remote Sensing, Vol 54 - No. 11, pp 1551-1555.
- McCormick, B.H., 1987. Visualization in Scientific Computing, ACSM SIGGRAPH, Vol 21 - No 6.

NATIONAL CAPITAL URBAN PLANNING PROJECT: DEVELOPMENT OF A THREE-DIMENSIONAL GIS MODEL

Lawrence G. Batten
U.S. Geological Survey
521 National Center
Reston, Virginia 22092

ABSTRACT

The U.S. Geological Survey (USGS), the National Capital Planning Commission (NCPC), and the U.S. Bureau of the Census (BOC) are cooperatively developing a three-dimensional model of the Monumental Core in Washington, D.C. The overall goals of the project are to extend two-dimensional GIS techniques into the third dimension and to illustrate the potentials of a three-dimensional GIS model for use in urban planning, review, and evaluation processes.

The major components of the study are the two-dimensional and three-dimensional models of the current urban setting including USGS Digital Line Graph (DLG), BOC Topologically Integrated Geographic Encoding and Reference (TIGER) system, and NCPC data sets; linkages to move attributes, spatial information, and analytical results between the two models; view-shed analyses of existing and proposed new buildings; and network analysis for urban transportation simulation modeling.

The ability to quickly and efficiently produce perspective plots from various view positions, to update the cartographic and (or) attribute data bases subsequent to design changes, and to model transportation patterns before and after construction of a new structure has made the planning review process more efficient and precise. New techniques developed during this project will also apply to the broader field of solids modeling including three-dimensional geologic, groundwater, and geophysical studies.

INTRODUCTION

Recent advances in computer technology have given researchers new tools for natural resource and socioeconomic analysis. Most notable of these advances are the development of workstations suitable for both three-dimensional Computer Aided Design (CAD) and two-dimensional Geographic Information System (GIS) software. Combining GIS and CAD technologies depicts urban and natural environments in a fashion more understandable to the lay person.

The following paper describes the creation of a three-dimensional data base for the Monumental Core of Washington, D.C. (fig. 1). The project is part of a

Any use of trade names and trademarks in this publication is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

Publication authorized by the Director, U.S. Geological Survey.

cooperative effort by the U.S. Geological Survey (USGS), National Capital Planning Commission (NCPC), and the U.S. Bureau of the Census (BOC) to develop computer tools for urban planning applications. Important considerations in the design of this project will be discussed and preliminary results shown.

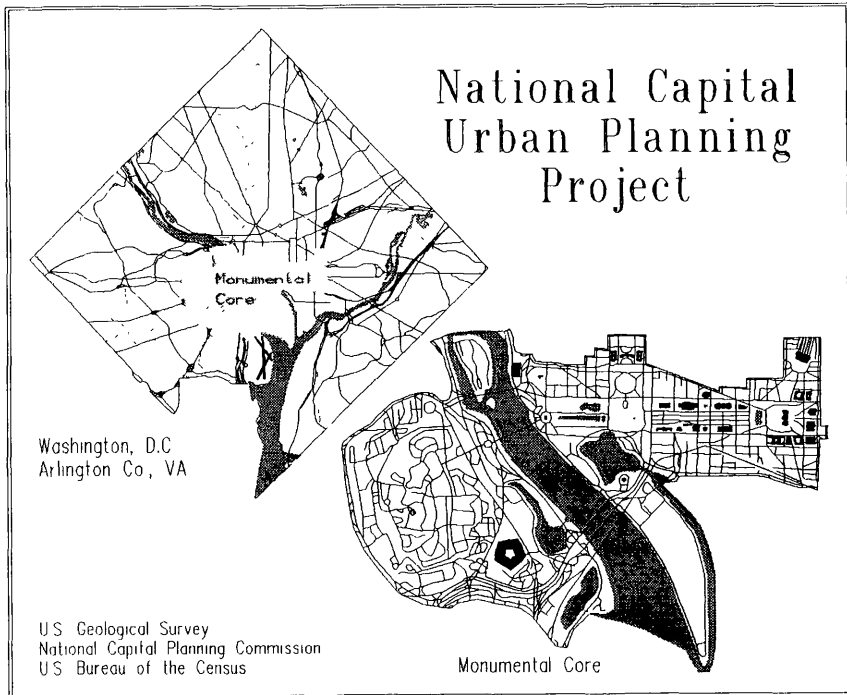


Figure 1.--The Monumental Core extends from the Capitol Building in Washington, D.C., to portions of Arlington County, Va., across the Potomac River, and encompasses major national monuments and landmarks such as the Lincoln Memorial, Jefferson Memorial, White House, and Pentagon.

BACKGROUND

As the central planning agency for the Federal Government in the National Capital region, the NCPC is responsible for the analysis of urban conditions, the review and approval of all new Federal buildings and projects in the Washington metropolitan area to ensure that they meet the long-range planning objectives of the Federal Government and are compatible with local plans, and the preparation of plans for future development. Detailed site analyses are produced for all proposed buildings to determine their effect on existing structures, public lands, and surrounding areas. Products include perspective views of the proposed building set in the midst of existing facilities, three-dimensional axonometric maps, and traditional maps at various scales and resolutions. Maps and graphics used in this process are currently being produced manually. The original data include USGS maps (normally 1:24,000 scale), aerial photographs, building construction documents, and local planning documents.

The USGS has developed numerous GIS data bases by combining disparate analog and digital data layers for subsequent modeling and program analysis. The USGS GIS Research Laboratory in Reston, Va., has vector and raster GIS software co-located in the same facility, which allows for the development of analytical techniques that utilize both storage systems. Hardware resources include three-dimensional graphic devices and GIS workstations.

The USGS and BOC are cooperatively preparing geographic data in support of the 1990 Decennial Census of Population and Housing. The resulting TIGER files offer a detailed description of the population for urban areas. To date, however, no GIS project has been applied to the study of the use of these data for urban planning. Both the cartographic and socioeconomic attribute portions of TIGER data are being used in the NCPC project to model this complex environment.

METHODOLOGY

The project was designed to develop three-dimensional GIS capabilities for use in architectural, urban design, socioeconomic and natural resource studies in an urban planning environment. A two-dimensional data base was prepared for traditional GIS applications and storage of detailed building attributes. A three-dimensional data base was constructed for maintaining structural details and producing perspective views. The development of two-dimensional to three-dimensional linkages for passing analytical parameters was a crucial part of the study.

Data Compilation

Review of the various types of data needed by the NCPC for the digital data base preceded the actual data collection tasks. Data included USGS digital topographic and cartographic data, BOC statistical and geographic data, urban planning data, and architectural drawings. The various scales and resolutions of the data were defined at this point. It was decided that the three-dimensional data base should contain enough detail to capture general building design but not to maintain specific details, such as ornamentation or brickwork. These fine details are more appropriate for a building-specific study rather than a citywide program.

Development of GIS Tools and Techniques

The second phase of the project involved development of the appropriate hardware/software environment for the two-dimensional and three-dimensional data bases. The NCPC and USGS had different equipment requirements because of their respective applications. A two-dimensional vector GIS system (ARC/INFO) was chosen by both groups to maintain the building-specific information such as number of employees, date of construction, and amount of office space. ARC/INFO was also used to generate cartographic products. Two three-dimensional CAD software packages were tested for solids modeling applications. The USGS installed Medusa on a Prime minicomputer, while NCPC used the Architect and Engineering Series software on an IBM RT personal computer.

Techniques for passing information between the two-dimensional and three-dimensional systems were written and implemented at this time. Unique feature codes were developed for all the structures within the project area. These codes were added to the two-dimensional data as an attribute in the INFO file. The same codes were then used as object identifiers in the three-dimensional data base. Typical GIS queries are then used to identify a subset of features that are of

interest. The codes for those features are passed as a file from the two-dimensional software to the three-dimensional data base where a perspective view can be generated showing those features in a different fashion. An example might be displaying a perspective view of buildings built in 1950 in red and the others in gray.

Data Base Development

After the proper GIS tools and techniques were determined, methods for entering nontraditional data (blueprints, building elevations, etc.) into the chosen GIS systems were developed. Manual digitizing was used to enter detailed data. Scan digitizing was used to enter data from planimetric maps and site plans. Rasterization of aerial photographs may be investigated later in the project.

The two-dimensional and three-dimensional data bases were treated as separate entities during this portion of the project. Each was different in the manner in which features were represented. For example, roads were entered as line coverages in the two-dimensional file and as polygons with x,y,z coordinates in the three-dimensional version. Creation of the two-dimensional data base followed well-defined routines from previous GIS studies; available two-dimensional analog and digital data, NCPC building attributes, and BOC TIGER files were registered to form a planimetric two-dimensional data base.

The major research product of this project was the creation of a three-dimensional model for the pilot area. The various buildings within the area were depicted as abstract geometric blocks or detailed structural entities depending on the application needs and graphics devices used. Most data were entered into the data base through turnkey operations available from the vendors. However, some computer code was written to enter data not routinely used in CAD systems (such as digital elevation data). Data sources included NCPC building "footprint" maps, low-altitude aerial photographs, digital elevation models, and building elevations.

Applications

Early in the project, a pilot area was chosen for testing modeling techniques. The site was representative of the urban design issues faced by NCPC. Accepted routines will be applied to the expanded area currently under investigation by the NCPC.

A proposed new Supreme Court building and associated changes within the pilot area offered the opportunity for real-world examples of the types of questions to be answered by the completed system. Perspective views of the study site before and after erection of the proposed new structure were identified as an important capability. The two-dimensional to three-dimensional linkages were crucial for querying the two-dimensional attributes and depicting the solutions in three-dimensional space.

Network analyses of the transportation environment within the pilot area were also conducted. Time-of-travel information was added to the attributes for the TIGER and DLG data. By using this variable, routes were computed to show the shortest travel time between various sites. Currently, methods are being developed for modeling the effects of increased commuter traffic from new buildings during rush-hours. Commute-to-work information maintained by NCPC and BOC will be used in this effort.

Socioeconomic analyses of the greater Washington, D.C., area are currently being conducted. Attribute data from the 1980 census have been combined with the

1990 TIGER files to produce a model depicting the cultural makeup of the region. Future applications of these data include cooperative studies with urban development, health, emergency, and law enforcement agencies.

Demonstration and Project Review

Demonstration and review of techniques and products developed are important aspects of the project. Presentations of the findings, techniques, and recommendations for follow-on studies will be made to NCPC and USGS staff and management representatives. Additionally, a number of Federal, State, and local agencies have shown an interest in this work for applications in urban planning, natural resource analysis, and data visualization.

CONCLUSIONS

The overall project goals were to:

- Design tools and techniques to assist the NCPC in planning the development of the National Capital region.
- Develop NCPC and USGS skills in the areas of three-dimensional analysis, graphics, and network studies.
- Illustrate the potentials of a three-dimensional GIS model for use in urban planning, review, and evaluation processes for cities throughout the Nation.

Project results demonstrate that using a combination of three-dimensional CAD and GIS is a better method for preparing the graphics used by the NCPC for their work. The ability to quickly and efficiently produce perspective plots from various view positions, update the cartographic and (or) attribute data bases subsequent to design changes, model transportation patterns before and after construction of a new structure, and enter engineering and design analyses (i.e., building height, infrastructure) will make the planning review process more efficient and precise.

This research has given the USGS, NCPC, and BOC experience in the new field of three-dimensional investigations. Results are more readily understood when viewed in a three-dimensional context. The new techniques developed during this project apply not only to architectural siting, but to the broader fields of solids modeling (including three-dimensional geologic, ground-water, and geophysical studies), view-shed analysis, and data visualization.

The new techniques also could be used by other cities as tools for urban planning. The ability to combine disparate data sets by using merged GIS and CAD technologies will provide researchers a better, more realistic view of the total urban environment. Future developments in automating data capture tasks, better transportation algorithms, and true three-dimensional graphic displays will further promote this technology for programmatic operations throughout the Nation.

**GIS FUTURE: AUTOMATED CARTOGRAPHY
OR GEO-RELATIONAL SOLID MODELLING?**

Hrvoje Lukatela
2320 Uxbridge Drive
Calgary, AB - T2N 3Z6 CANADA
(Envoy 100: lukatela)

ABSTRACT

Computer Mapping and Geographic Information Systems have evolved into two distinct computer application areas. While the two share some common ground, their differences are significant enough to merit an independent search for the basic software design strategy.

This paper concentrates on elements which characterize and define Geographic Information Systems - in contrast to Computer Mapping. It then explores their influence on system design criteria and software engineering aspects, such as the data modelling, spatial operators, external storage management, reference surface selection and computational geometry.

PURPOSE

Due primarily to the reasons of history and technical tradition, Geographic Information Systems are routinely built using the software design fundamentals originating from the much older field of Computer Mapping. The purpose of this paper is to contribute to the discussion of alternatives, which could result in better designed software and higher functionality of such systems.

Issues are examined strictly from the technical perspective of a system designer: operational value of the system will, in addition, depend on the organizational issues, which can be analyzed according to similar criteria.

DEFINITIONS

Computer Mapping Systems (CMS) are applications which store and manipulate location-defining attributes of data objects, with the purpose of generating their analog, graphical representations. Those can be either permanent or transient, and can emphasize spatial relationships between objects ("overlays" etc.) or include graphical portrayal of additional, non-spatial data attributes in spatial context ("thematic cartography"); but their utility is restricted to the visual consumption by a human operator or system user.

Geographic Information Systems (GIS) are applications based on a data model dominated by the location-defining attributes of its objects, capable of data processing required by an administrative, technical, educational or other discipline, in order to automate some of its functions and processes. While the generation of analog view of data can be (and almost always is!) included in the functional repertoire of the system, it is not its primary function. Indeed, GIS often use a combination of location-defining and other attributes in processes

which mimic application-domain inferences, procedures or depictions, and produce results which are output not in cartographic form, but in the form native to the application itself (e.g. report, table, decision selection, statistical graph or control loop signal).

In short, Computer Mapping Systems automate the process of composition and production of analog map documents; Geographic Information Systems use "digital maps" in order to perform functions intrinsic not to the cartography, but to some practical discipline that studies geographically distributed data.

DATA MODELLING AND SPATIAL OBJECT CLASSES

CMS data models are usually based directly on the graphical scheme that is at the same time a precise description of the system output. It is typically restricted to 0 and 1 dimensional objects, (points and lines), and includes attributes which specify the details of graphical presentation (symbols, colour, line style, label placement, etc.). The system can also include non-spatial attributes, or be partitioned into pre-defined "layers", representing different classes of spatial objects.

Since GIS are primarily application systems serving an unending array of industries and disciplines, "GIS data model" can not be addressed in a generic form. Generalized, application-independent theory of data modelling enjoys currently the research attention beyond anything that a specialized field like a GIS can convoke. Results are directly applicable to GIS objects which are not spatially-defined, and also to the non-spatial attributes of spatially defined objects.

The geometry of spatially defined real-life objects will usually be significantly more complex than the geometry of a CMS graphical scheme. This will be caused - typically - by a combination of more complex geometry, and by the temporal nature of the object shape, size or location. It is in this area that GIS requires specialized modelling techniques.

Invariably, practical system design considerations will require some simplification as a part of the process of transformation from the object into its digital representation. The central problem of the spatial data modelling consists therefore of establishing the balance between the simplicity and faithfulness of the spatial object representation: overly complex representation will make the system too costly to construct and operate; insufficient faithfulness will impair the functional power of the system and thus reduce the benefits of its implementation.

While many CMS data models are built using only simple point and line objects, GIS are usually required to model objects with more complex spatial or spatially-temporal definition. Those listed below - in the order of increased complexity - probably represent the most common classes of spatial objects stored on GIS:

- 1) **Point set:** a finite number of surface point locations. (The set is aggregate, and all non-spatial attributes pertain equally to all locations in the set.)
- 2) **Discrete surface value set:** point set with a single, numerical value associated with each location in the set.

- 3) **Line set:** one or more ordered point sets.
- 4) **Gravitational movement trace:** parameters defining geometry of the conic section, its external orientation parameters in the global frame of reference, and a point on the curve at the epoch.
- 5) **Kinematic movement trace:** ordered series of surface point locations (possibly including normal displacement coordinate), with time value associated with each.
- 6) **Surface region:** one or more ordered point sets, representing boundary rings of a non-simply connected surface region. (Boundary must not cross itself, and ring directions must be consistent among all the rings in the set.)
- 7) **Boundary system:** a composite object consisting of an ordered list of single-location point sets representing the node points and an ordered set of line sets representing the boundary segments. (A series of boundary-system-object/node-point/segment identifiers can be used as an alternate spatial definition of a single-ring region object.)

GIS spatial object lists, such as the one above, are by nature open-ended. A complete list can only be composed based upon a valid data model for a specific application. The designer of application-independent GIS software tool must, however, take a more pragmatic approach: a finite collection of objects must be selected and implemented as intrinsic to the package. The application builder can then restrict his data model to the objects supported by the tool, or extend it by providing additional data structures and functions in the application software.

Either the tool - if it is used - or the application code must provide a complete set of spatial operators, capable of deriving spatial unions and intersections between all union-compatible pairs of object classes. In CMS systems such operators are used on their own, and the results of their invocation are displayed graphically. In GIS, spatial operators are frequently combined with processing based on non-spatial attributes in complex, response-time critical transactions, which do not tolerate relatively low level of efficiency of spatial operators commonly found in CMS. In addition, such transactions can create new spatially-defined objects, which the system must be able to treat in exactly the same way as source objects - e.g. display, store on the data base, export to another system etc.

Different objects will in general be spatially defined with different levels of precision, and their representation should take this into account by using different internal coordinate item width. Since this can be provided only in discrete steps (e.g. single and double precision floating point representation of coordinates) each object must carry an item which indicates, numerically, spatial precision (as opposed to the artificial data resolution) with which the object is known to the system. It is worth noting that the absence of such indicator will influence CMS only in a limited manner: once scaled down to the size of its graphical depiction, precision related problems will be insignificant compared to the same in GIS, where spatial relationships are evaluated in object-space size and precision.

The level of spatial precision required for various data objects depends on the target system output precision, but also on the data model characteristics:

As long as the performance is not critically affected, geometry elements that can be derived from the primitive location attributes should not be stored on the system, but derived as, and when, required. However, primitive attribute precision requirements can often be relaxed, if the precision-critical elements are stored redundantly. (Common examples include precise land-parcel areas, or distances along the centre-line of a liner feature, stored explicitly, in a combination with point coordinates scaled from the map.) Such inconsistencies in the geometry model can severely restrict implementation of functions which have not been "built-into" the original design; principles of "open-ended" system design influence equally the spatial and non-spatial elements of the data model.

EXTERNAL STORAGE MANAGEMENT

One of the common characteristics of CMS and GIS is the high data-volume brought about primarily by the nature of location defining attributes. The data retrieval patterns, however, are different. In CMS, most partial retrievals will be geographical, limited to the current location of the display window. Special-purpose storage indexing schemes (based mostly on various algorithmic representations of regular planar tessellation - e.g. B-trees, Quad-trees, R-trees etc.) have been both well researched, and verified in practice in many generic CMS packages. Variety of objects stored on a CMS data base is usually restricted to a relatively small number of fixed "layers" or "record types". Comparatively low volatility of CMS data bases makes system implementors and operators relatively undemanding in the areas such as on-line update transaction support, access control, ease of backup/restore process, encryption, multiple update conflict resolution, and a large number of other facilities considered sine qua non in a modern data base software package.

GIS data bases parallel those of most large data base development projects, but must, additionally, allow spatially-defined retrievals.

Current preponderance of the relational data base model has significant repercussions on the whole GIS realm. Both the application-domain expert and the application programmer are likely to demand and expect the flexibility and conceptual simplicity associated with the relational model on both the non-spatial and spatial data base development projects. From their viewpoint - when it comes to manipulations performed by the data base manager software - there must be little or no difference between the spatial and non-spatial attributes of their objects.

If that is the case, spatial retrieval schemes and attribute storage must follow the general philosophy of the relational data base model in several important aspects:

- Spatial relationships must not be encoded with the data (in form of pointers, "topology indicators", spatial structure descriptors, etc.), but must be derived from the location-defining data attributes at the time of retrieval processing.

- Spatial retrievals must be possible based not only on the pre-defined surface elements, but also on relational algebra productions between objects on the data base.
- Introduction of spatial criterion into a complex retrieval selection set must not complicate the retrieval request formulation more than would be the case if an additional non-spatial criterion was introduced.
- Redundancy criteria and degree of normalization of spatial attributes must equal those applied to non-spatial attributes.

GIS data compilation that does not violate above principles can be called **"geo-relational"** data base. Both the application programmer and end-user alike will perceive it as a relational data base in which spatial and non-spatial attributes have been integrated in a seamless manner.

Since few projects can justify dedicated efforts required to implement complete data base manager software, GIS system designer has two alternatives: a generic CMS package which provides some degree of non-spatial attribute support, or a general-purpose data base package with the addition of functions providing spatial data storage and retrieval. It appears that at present the former alternative enjoys greater popularity among GIS system developers. This might change: emphasis on the relational model and associated flexibility of the data base design, combined with the increased demands on the operational strength of the data base, makes current generation of data base products very attractive to the GIS implementor. This might, in turn, provoke the emergence of software products which will provide the necessary set of geo-relational functions in the form of specific data base manager "add-on" packages.

REFERENCE SURFACE AND COMPUTATIONAL GEOMETRY

Cartographic projection plane is the spatial data domain of most CMS. This is not the case with GIS: even when coordinates used as location defining attributes are e.g. Universal Transverse Mercator (UTM) projection plane "northings" and "eastings", they conceptually represent locations on the surface of the Earth. This becomes obvious when the need arises to model an object which extends across two different UTM projection planes: the object itself (unlike parts of its depiction) does not belong to two distinct data domains! In GIS, cartographic transformations - if used at all - only "mask" the spatial data domain by defining it implicitly by the way of their own parameters and algorithms.

True spatial data domain of GIS is always part (or whole) of the planetary (or "reference") surface, or, in different words - **reference surface is the dominant spatial object of every GIS.**

In order to formulate computational geometry - a set of rules used as a base for derivation of spatial relationships between the objects - the reference surface must be defined in a simple, yet sufficiently precise form. Plane, sphere and two-axial ellipsoid are commonly used for this purpose. (Since the reference surface interacts with every other spatial object, more complex reference surface definitions are of little or no practical value to a GIS designer.) The simplicity-

-precision scale is obvious: plane is the simplest, and ellipsoid the most precise GIS reference surface.

Before the question of "sufficient precision level" is examined, it is important to note that - by definition - the computational geometry used (planar, spherical, or ellipsoidal) must yield all numerical results with the level of precision required by the system, without abandoning the metrics implied by its definition. As an example, if the system uses UTM coordinates to define locations, but "corrects" coordinate diagonal (by applying UTM "scale factor") to obtain distances between two points, its computational geometry (and therefore the reference surface) are clearly not planar, but ellipsoidal, defined implicitly by the correspondence of UTM and ellipsoid coordinates.

As mentioned before, all objects modelled by the system will not be known with the same spatial precision - or represented numerically with the same resolution. The inaccuracies introduced by the spatial frame of reference must not lower the accuracy of the most precise derived data item generated by the system. In general, this will be achieved if the distortions introduced by the geometry of the reference surface are kept at least a full order of magnitude below the resolution of highest-precision data items.

Few - if any - GIS can be constructed using the plane as the reference surface, without violating this principle. (As an example, a simple municipal cadastral and engineering data base, extending over an area with a radius of only 20 Km, with coordinates defined to the precision of .1 m can not be constructed in plane, if the geometry of objects larger than some 250 meters in diameter is to be generated from the coordinate data!) Planar cartographic projections are therefore of little value as GIS reference surfaces; their use should be restricted to the necessary conversion of coordinate data on output and input from and into GIS.

Spherical models - particularly those based on spherical coordinates obtained by either rigid or approximate orthomorphic transformation from the ellipsoid - are much more likely to satisfy GIS reference surface precision requirements. Using the same municipal system example as above, the radius of the area of coverage would have to extend more than ten-fold before the same distortion is encountered. Another example: single-plane data domain GIS covering the contiguous United States are commonly constructed using a "two-parallel" Lambert projections. Maximum (local!) linear distortion of such system will be as high as 1 in 50. Local linear distortion of the same system based on a single orthomorphic sphere would be only 1 in 1500 - accuracy level almost approaching that of a single UTM plane system (1 in 2500). In addition, spherical computational geometry is based on simple, closed numerical relationships, which, compared to planar calculations, require no more than a modest (typically in the order of 50%) linear increase in computing power.

Large-area GIS, particularly those that include numerical data representing distance or direction measurements carried out at object-scale, may require ellipsoid reference surface in order to attain the required spatial precision level. The ellipsoidal computational geometry presents the algorithm designer with a significantly more complex series of problems, and may require dramatic increase in computing power.

While such calculations are still practical in case of low data-volume problem solutions, high data-volume problems - as, for instance, evaluation of spatial unions and intersections - will require better techniques than those which are considered satisfactory for planar or spherical systems.

One class of GIS makes use of ellipsoidal spatial reference system almost mandatory: systems that relate data acquired from orbiting sensors and terrestrial data originating from a large variety of conventional sources. (Present stratagem of pre-casting the sensor data into some particular projection plane geometry, scale and pixel-ratio is satisfactory as a base of CMS, but of little or no usefulness for GIS.) Manipulation of high-density sensor data will require techniques similar to those necessary for the evaluation of spatial unions and intersections. Once developed, such techniques will make possible the solution of complex, high data-volume spatial/terrestrial problems in the most logical frame of reference for their solution: one based directly on the metrics generated by the same force that shapes the terrestrial surface, and determines the trajectory of a free-falling sensor: the inverse r^2 force!

CONCLUSION

The primary design problem of most Geographic Information Systems is that of integration of two classes of data and procedures: those that define spatial characteristics of objects, and those that describe complex, application-specific qualities and measures of same objects.

Digital modelling of spatial objects in GIS must be optimized toward object-space precise and efficient evaluation of complex relationships defined by a combination of spatial and non-spatial criteria; numerical models developed for the purpose of either manual or computerized map production are not adequate for this purpose.

SPECTRAL/SPATIAL EXPLOITATION OF DIGITAL RASTER GRAPHIC MAP PRODUCTS FOR IMPROVED DATA EXTRACTION

Timothy J. Eveleigh, Kevin D. Potter,
Autometric Inc., 5301 Shawnee Rd.,
Alexandria, VA 22312

ABSTRACT

In recent years, much research and development has been successfully applied to using image processing technology to exploit multispectral scanner remote sensing products for geographic information. Recent developments in technology have made affordable scanners that can allow systems as small as home computers the ability to scan-digitize map products through broad spectral filters. The resulting digital raster graphics (DRG) are also becoming increasingly available commercially. The spectral and spatial nature of these products shows great potential as a source of exploitable geographic data. The same image processing techniques now applied to remotely sensed imagery can in many cases be applied to DRG. The advantage of such an approach is that it can speed data extraction while performing data reduction. By performing spectral classification of the DRG and intelligently assembling thematic components of the resulting classified image, items such as vegetation and water can be extracted. Further processing could extract linear features like boundaries, roads, and even contour lines. These linear features have the potential to generate geographic information like road networks and DTMs. This paper presents the preliminary results of an investigation into the spectral and spatial exploitation of DRG using image processing technology.

INTRODUCTION

Since the early 1970's, the extraction of geographic data from multispectral imagery products has evolved from a laboratory curiosity to an essential tool. Classifying multispectral imagery into discrete spectral classes that can be correlated with geographic knowledge and subsequently managed by a geographic information system is an effective tool used widely by resource managers.

Recently, advances in computer technology have made affordable powerful small computers capable of supporting image processing software and geographic information systems. Optical scanners which can convert hardcopy products like maps to digital images have also become more available to the low end user. Traditionally at the disadvantage of being memory intensive, the digital map products have seen limited use as analysis backdrops and to support manual softcopy digitizing of geographic features. Developers of memory technology, however, are continually introducing more capable, higher volume, cost effective data storage solutions for small computers.

In addition to the increased accessibility of hardware, an increase in the availability of digital map data is imminent. This is due in part to the increased availability of low cost scanners. It is also due to the map data production community's intention to introduce the widescale production and dissemination of cartographic products in digital raster graphic (DRG) form.

In light of the potential increased availability of DRG, and that DRG is already resident in a form that a computer can manage, it is useful to consider how scanned maps can most effectively be exploited. For resource managers and users of geographic information systems (GIS), one of the most desirable uses of scanned maps would be to employ them as a source of data to rapidly populate a geographic database. It would be very desirable to feed a scanned map to a process that would interpret the map and create a topologically valid geographic information model that described the mapped terrain. Such a tool could perhaps even obviate the labor intensive manual digitizing and attribution process commonly applied to hardcopy and softcopy products. Although such a technique would not correct map errors and would retain any biases or defects in the map product, it could provide a rapid 'first cut' that would be suitable for many applications.

Such a tool would have to be able to discriminate individual colors, patterns, and shapes on the map product. The tool would have to be told the ranges and manifestations of map data and what the geographic features represent on the earth's surface. This is a considerably involved task and in many respects is similar to automated image understanding. This paper presents some preliminary findings of applying to raster scanned map products existing and emerging technology developed to support image understanding.

SPECTRAL EXPLOITATION OF DRG

Scanners used to convert hardcopy products to a digital form are similar to the multispectral scanners employed by earth imaging aircraft and space platforms. Both instruments collect a grid of samples of an objective through optics, pass the incoming radiation through bandpass filters, convert the filtered radiation levels to electrical charges, digitize the electrical charges, and store the digital data in a form that can be reconstructed to produce brightness images for each of the filtered bands. Table 1 shows some of the similarities and differences between the scanners and data from the two systems.

If the two systems generate images that share many characteristics, it would follow that traditional multispectral analysis could be applied to the DRG. On first inspection, it would appear that the application of traditional multispectral analysis to DRG is a much simplified case of classifying remotely sensed imagery of the earth's surface. Due to the generalized and symbolic

Category	Traditional Remotely Sensed Imagery	Digital Raster Graphics of Maps
SPATIAL SAMPLE RESOLUTION	Usually fixed and determined by platform.	Most scanners capable of several.
SPECTRAL BANDS	Can be many, determined by choice of platform.	Rarely more than three, determined by hardware.
REGISTRATION TO GEOGRAPHICS	Requires registration via rigorous sensor model and/or extractable control points.	Usually retains the projection and tick marks of original map therefore can be modeled using map registration software.
SPECTRAL QUALITY	High spectral quality due to calibration and design of sensor payload and knowledge of illumination conditions.	Depends on machine used and how operated.
SPATIAL QUALITY	High spatial quality due to precise knowledge of data collection activity.	Depends on machine used and how operated. Contiguous patches may not join up.
APPEARANCE OF FEATURES	Imaged features are as they actually appear in location, size, and shape.	Imaged features are often symbols that differ in size, shape, and sometimes location of actual items portrayed.

Table 1

nature of maps, there should be far fewer spectral clusters for the computer to resolve.

Computer extraction of geographically significant spectral classes or colors from the printed map surface can be complicated by several factors, however. The first such factor is the aliasing inherent in the sampling approach used by scanners.

When the human eye reads a map, the mind automatically performs several functions that a computer might not ordinarily perform. For example, most printed maps contain colors and shades of colors that have been printed with dot screens. A small light blue lake, for example, is actually printed as several thousand blue dots. The size and distribution of the printed dots will determine how the brain processes and classifies the overall color of the stippled area. A computer, however, if fed a buffer of pixels that contains the blue dot field will not reclassify all the pixels near the blue dots and the dots themselves as another shade of blue, it will continue to handle each pixel as a discrete element - unless it is instructed to do otherwise. The computer will simply classify the blue pixels as lake, and the white background as not lake.

The literal translation of dot patterns begins to approach the desired result of seeing the lake 'color' if optical mixing occurs during the scanning step. If the resolution element size of the scanner is increased out of the size range of elements of the printed dot screen pattern, the white and blue contributions to each pixel begin to mix producing a set of light blue colored 'mixels'. These mixed contribution pixels can be spectrally identified and used in the classification stage. The sample interval will

determine how many mixture classes there will be for a given screened pattern. Choice of the scanner's sample resolution is important, as it not only affects data storage requirements, but it will determine the number and nature of the spectral classes that indicate map colors (and therefore, the complexity of the data classification and extraction process). Pixel mixing will also occur along edges of features resulting in the generation of several new mixed boundary classes.

Due to all the sample mixing, successfully defining the spectral representation of a screened pattern like a lake begins to approach the complexity of classifying an actual lake on remotely sensed imagery. Instead of a sharp, low spectral diversity cluster, the map lake color definition becomes the product of the many pixel mixes that will occur when the water pattern is sampled. Extraction, however, is still possible as long as the contributing clusters are mathematically resolvable within the set of all scanned map reflectances. We find that in many instances, this is the case for DRG map data, as it is for multispectral remote sensing imagery.

Another factor that can interfere with the raster map to thematic map conversion is that many of the geographic features depicted on a map are represented by cartographic symbols that mean nothing to the computer. Features like structures, roads, political boundaries, text, tick marks, and contours are all represented by a mix of linear and point symbols. Often these symbols accurately depict the location of geographic features, but distort the dimensionality of the features. Spectral classification of the map product could potentially locate these symbols, but could not convert them to appropriate geographic analogues. These items will require an additional spatial analysis or even pattern recognition step before being thematically attributed.

RESULTS

By applying existing multispectral image classification software to DRG maps, we were able to explore the process of spectrally classifying the map images into geographically significant cell maps. We found that by using a supervised spectral training approach, the majority of shaded and solid map color spectral clusters could be determined, and that these clusters when fed to a maximum likelihood classifier resulted in useable thematic geographic information. We also found that the process requires a large amount of guidance from the operator (or as we propose, a knowledge-base system). Guidance was required both in the spectral training phase and in the conversion of the classified cell map from a 'color' map to a geographically significant thematic map. The conversion of the 'color' map to the geographically significant thematic map is where we found the most significant departure from the remotely sensed imagery to geographic data conversion.

Let's take the example of the task of extracting forested

areas from imagery and from DRG maps.

A forest as seen on an image will manifest itself as a complex cluster or set of clusters in spectral space. Many factors will contribute to the total range of spectral space caused by the interception and filtering of energy reflected by the forest's canopy into the sensor's optics. To classify accurately the entire image into classes of forest and not forest, one would hope to describe this complex spectral distribution as completely as possible before or during classification. Groundtruth permitting, one would have the option of attempting to establish additional subclasses of forest based on measurable or known differences in canopy reflection between tree types. Following the definition of the forest's spectral space definition(s), the entire image would be subjected to classification and a raster map of forest distribution would result.

On a simple map, however, the forest could appear to the eye only as a solid shade of green. Were this the case, classification and extraction would be simplified. The green reflectance of the map should hold nearly uniform assuming that the illumination of the map during scanning is uniform. The dot spacing of the screen used for the green printing should also hold constant across the map. In this simplified case, the delineation of the spectral space definition of the forest should itself be simple. Once the discrete signatures of the important colors (green and not green) are delineated, the classification step should provide a raster map of green color distribution which we will call forest. Unless individual forest types were symbolized by the map maker, however, the ability to distinguish forest types as classes is lost.

The simple map case is rapidly complicated when actual maps are used. First, we have the aliasing effects introduced by the dot pattern and forest edges that we mentioned earlier. We also encounter another pixel mixing effect - this time caused by overlapping map colors. The overprinting of cartographic elements such as text, symbols, and contours alters what must be identified as the forest's spectral space. For example, the overprinting of a brown contour on a green forest color can result in several new discrete spectral clusters. In this case, the spectral definition of a forest on a map must be amended to include contours within the forest, otherwise the classified raster forest map might not include pixels containing contours (see Figure 1).

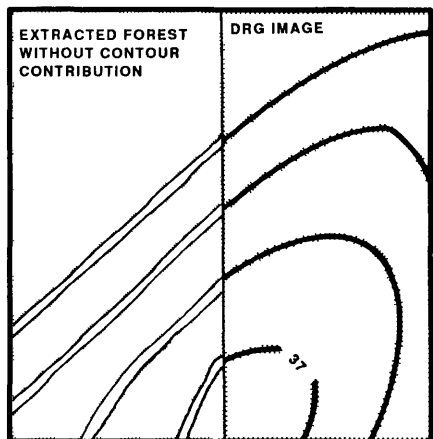


Figure 1

This is also true for other mixtures like political boundary demarcations, colored text, and relief shading. What began as a simple hunt for light green pixels can end as a hunt for pixels in a variety of colors, and a need to combine intelligently many of the classes from the output map to assemble the entire 'forest' distribution. For example, the 'forest' class might end up being the combination of the following resolved classes: Mid green shade + edge green shade 1 + edge green shade 2 + brown and green + edge brown and green + brown and blue.

Let's examine the case of the extraction of forests from DRG maps further. For example, it is entirely possible that forests might not be the only features represented with green ink. If the above technique encounters an orchard for example, the resulting classified raster map will contain blobs of forest where the green orchard symbols were encountered. Although this is not always undesirable, the exact boundaries of the orchard are lost, and potentially incorrect positions for trees are retained (see Figure 2). Orchards, scrub, and vineyards could all fall prey to this problem if one were extracting from a chart that represented these features with a pattern versus an individual color.

Certain patterns can also introduce additional aliasing issues. Consider if you will a symbol or shading pattern that contains closely spaced stripes, dots, or cross hatching. If the sampling interval of the scanner was smaller than the width of the individual dots, stripes, or cross hatches, they would be converted to the target class. If, however, the sample size increases, the potential for interference colors increases. Interference colors are not a problem if they are expected, and if they are accounted for.

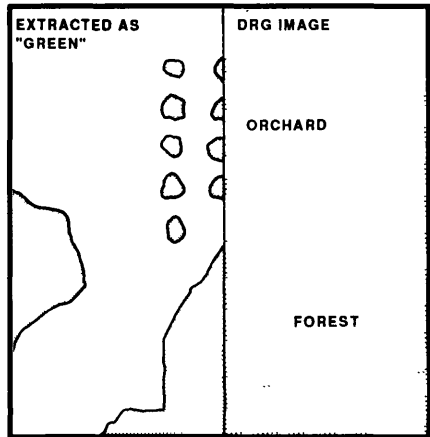


Figure 2

SPATIAL PROCESSING

Spectral processing of DRG can be used to provide a classified cell map representing the discrete colors of a digital raster graphic as seen by a scanner. Many shaded polygonal map features like forests and water bodies can be assembled out of the classified colors that result. Some polygon features, and most linear features, however, will need additional spatial processing to more correctly represent the geographic items depicted. Tools like line following algorithms can be brought into play to help achieve this end.

A good example of the need for such processing is provided by an attempt to create an arc-node contour map from a raster scanned portion of an USGS quadrangle.

The first stage in this study was to identify the discrete spectral signatures that represented the various colors that the brown contour lines make as seen by the scanner. We found that two discrete signatures defined ninety percent of the brown color used to denote the contours. These two colors were mainly the result of additive mixing of the contour color with the color that it overlapped. Contour (brown) on background (white) and contour (brown) on vegetation (green) were the two dominant signatures. The remaining ten percent of the contour pixels fell into a series of signatures that represented various combinations of green and brown, white and brown, brown and blue (at stream crossings), and brown and black (at road crossings). In an attempt to define areas that were not contour, pure black, solid green, solid red, solid blue, and solid white signatures were measured. When maximum likelihood classification was applied to the entire map using all of the above signatures, a raw raster contour map was produced.

Observing the spectrally classified map, it was obvious that many segments of contour pixels could be used by line following software to create clean linear features. Furthermore, when these linear features are correctly linked and attributed, the resulting vector map could be used to generate a rough terrain model.

Other segments of contours fell victim to the choice of scanner sample size. Problems arose when two or more contour lines were in close enough proximity to occupy the same or adjacent sample pixels. This problem manifested itself as contours that appeared to melt into each other. In areas of high slope this could present a significant challenge to contour extraction and editing software. One solution to this problem would be to increase the density of the scanner's sampling. This approach would allow the technique to reduce or discard many of the mixed signatures that fringe the contour lines, and to resolve smaller gaps between printed contours. The disadvantage of this approach would be an increased cost due to increased data storage requirements and the time spent processing the map.

A few contours also contained breaks with numbers indicating the elevation of the contour. The classification technique classified these numbers as contours as they were printed in brown. In our approach, we will attempt to filter out these discontinuities during processing of the contour data, though potentially they could be automatically exploited by pattern recognition software and used to assign a quantitative attribute to the linear vectors being assembled.

The processing of streams and linear cultural features like roads, rails, political boundaries, and map tick marks could follow a process like the contour extraction example. A big difference here is that the symbology used to depict these

features is far more varied in color, thickness, and shape compared to the relatively easy to extract contours. Dedicated software could be written to extract each of these entities.

Point features also have a potential for extraction. Like the above mentioned polygonal and linear extraction examples, it is possible to extract classes of color used to depict point features. The extracted features would not appear as points however, they would appear as the shape of the symbol used to represent the point features. Here too, it is conceivable that pattern recognition technology could be brought into play to convert the cluster of pixels into an attributed point feature.

FUTURE CONSIDERATIONS

During our study of applying a remote sensing image data extraction approach to DRG maps, we found that several of the existing tools (multispectral signature measurement and management, maximum likelihood spectral classification) could be effectively applied to DRG to create a cell maps that represented unique combinations of map colors. We also noticed that this approach generated several mixed colors resulting from optical mixing during the scanning process, and the overprinting of map colors. Following color classification, available software allowed us to combine classes to form cell maps that were geographically relevant or that could be made geographically relevant by additional spatial processing. What was missing, however, was any guidance that could be used to direct the task of extracting a given feature category from a given DRG. Such guidance would have to be available if the technique were to be automated. This is where we believe a knowledge base system (KBS) could be applied.

A knowledge base system establishes and administers a set of rules and procedures for accomplishing tasks. For example, a knowledge base system could provide a mechanism for directing the extraction of a particular geographic feature type from a given DRG source. The knowledge base would direct the operator or spectral clustering algorithm to collect the spectral signatures necessary to define all of the color manifestations of the feature(s) being extracted. The knowledge base system would then be called on to assemble intelligently the thematic cell map from the jumble of available cell classes following the classification step. Finally, the knowledge base system would direct the assembled map to any appropriate spatial processing algorithms for further extraction.

A knowledge base system can be used to make the classification algorithms more efficient by limiting the total number of mixed color signatures to the minimum required. The knowledge based system performs this task by pruning irrelevant signatures.

An important feature of a knowledge base system is that it is heuristic and can be trained to respond to a variety of

applications and source data types. We believe that a properly trained knowledge base system can be effectively employed to automate most or all the phases of geographic feature extractions from DRG. This capability would greatly enhance the utility of DRG products by making them sources of geographic information.

DATA AND EQUIPMENT USED

Our study examined sections of a rural USGS 7.5 minute quadrangle map that were scan digitized into three 8 bit images (red, green, and blue). A HOWTEK Scanmaster tabletop scanner was used to raster digitize the map product. A 200 sample per map inch resolution was used. We employed a Compaq 386-based version of the ERDAS image processing package to perform our experimentation with the DRG's spectral nature and spectral reclassification. We employed the capabilities of ERDAS' cell file management software (GIS) to perform class reassembly.

CONCLUSION

Recent developments in technology will soon unleash a flood of affordable rasterized map products. These digital raster graphic map products have a great potential as a source of geographic information. Moreover, this information could be extracted in a semi-automatic manner. The ability to perform automated geographic data extraction from DRG products will increase their value to managers and collectors of geographic information, and could eventually reduce the labor intensive step of manual data extraction.

We have examined how off-the-shelf remote sensing image processing software can be employed to extract the basic colors and patterns from a map surface necessary to build geographically relevant information. We have also experimented with the guided recombination of the spectrally classified cell data and examined some of the factors that contribute to the generation of the cell classes. It remains to be seen how successfully we can apply spatial analysis tools to further develop the geographic information that results from class assemblage. Similarly, it remains to be seen how automated entire feature extraction scenarios can be made when a rule base system is employed to control the process. We believe that both approaches have the potential for extracting a variety of feature types from digital raster graphics. It is important to note, however, that the best-case extraction possible from a DRG product will only be as accurate as the map product used as a source.

CARTOGRAPHIC DATA CAPTURE USING CAD

Michael E. Hodgson
Department of Geography
University of Colorado
Boulder, CO 80309

Ann L. Barrett
4361 Butler Circle
Boulder, CO 80303

Reese W. Plews
Department of Geology and Geography
Hunter College - CUNY
New York, NY 10021

ABSTRACT

The digitizing of cartographic features is a necessary but laborious task to many research analysts undertaking cartographic or GIS studies. The requirements of a good digitizing module for the capture of such cartographic features are discussed in this paper. To meet these requirements, the advantages and disadvantages of using an existing cartographic digitizing module or a CAD package are discussed. As a CAD package cannot meet all of the requirements, the design of a CAD post-processing program for assembling polygons from chains and for automatically relating attributes to objects is presented.

INTRODUCTION

The manual digitizing of cartographic data (i.e., with a puck and tablet) is an important process in many applications of digital cartography and geographic information systems (GIS). The availability of digital cartographic data from federal, state, or private distributors has provided an enormous supply of existing digital data set for such applications. However, the manual capture of cartographic data will continue to be a necessity for all but the most widely used maps and study regions that may be available in digital form. The capture of such data is hampered due to the availability/cost of good capture software, hardware requirements/cost, and the digitizing time required by the analyst. A variety of capture software packages are available; however, most packages are either expensive, cumbersome to use, have limitations on map complexity, or only create output formats for a limited number of cartographic and GIS packages.

This paper describes the requirements for a good geographic data capture package and the abilities of a computer aided design (CAD) package to meet these requirements. The logic and specific commands for using the AutoCAD package for recording geographic data are described. The design of a post-processing program used to build polygons and relate attributes to cartographic objects from an AutoCAD file is presented.

REQUIREMENTS FOR A GOOD GEOGRAPHIC DATA CAPTURE PACKAGE

Before discussing the requirements for a good cartographic data capture package, the fundamental definitions of mapped features as related to this study must be presented. All geographic features on a map may be recorded as either point, line, or areal objects (assuming that a surface may be recorded as a set of points or isolines with an elevation attribute.) In most digital cartographic data structures, point, line, and areal features may simply be defined by their X and Y coordinates for location and one identification attribute for further describing the object. This attribute may be used to describe the general class that the object represents (e.g., 11 for residential areas) or to describe the unique individual object (e.g., a 5-digit FIPS code for a U.S. county). Further, other attribute values may be "linked" to each object using the unique identification attribute and a relational database.

According to the proposed standards of digital cartographic data, the digital representation of entities on a map are *objects* and may be further defined as a variety of specific 0, 1, or 2-dimensional objects (DCDSTF, 1988). The appropriate objects used in this study are: entity and label points, nodes, strings, chains, rings, and polygons. An *entity point* is a set of coordinates defining the location of a point feature (e.g., tower, building). A *label point* is a set of coordinates used for locating text on a map for feature identification and is used in this study to describe the identification attribute. A topological junction or endpoint (e.g., junction of two or more linear features) that may also specify location is a *node*. A sequence of line segments (without nodes or left and right identifiers) that may intersect itself is a *string* and will be used for recording linear features. The boundaries between areas will be recorded as *chains* - a directed sequence of nonintersecting line segments. The areal features on a map are defined as *simple* or *complex polygons*, depending on the absence or presence of inner rings (i.e., islands or enclaves). Each polygon is further described by one or more *rings*, which may be composed of a sequence of chains that represents a closed boundary around the area defined.

To record these mapped features, a good digitizing package must allow string-node or chain-node digitizing capabilities for polygon capture (formerly referred to as arc-node) and the automatic "relating" of attributes for each object. String-node or chain-node digitizing of polygonal objects is faster and eliminates the sliver lines between adjacent polygons. The relation of an identification attribute for an object (such as a FIPS code for a county) should be automatically performed. The package should also have error detection and display capabilities, be easy to edit, and be easy to learn and use. Ideally, the package supports color displays, is inexpensive, allows unlimited map complexity (e.g., number of polygons, points in a string, etc.), and creates output to all mapping and GIS packages.

ALTERNATIVES FOR DATA CAPTURE

Advantages and Disadvantages of Available Digitizing Modules

Many mapping packages and GIS packages have a cartographic data capture module included as part of the package. The data capture module is designed to work in an integrated environment with the other mapping or GIS modules. The data capture module for a software package will produce an output format compatible with the package which it is a part of. Unfortunately, digitizing modules rarely create output formats for the competing industry products; thus, the development of data interchange software is required or even the procurement, training, and use of more than one data capture modules is necessary. Further, many digitizing modules only operate on a small number of hardware configurations. The lack of available data capture software and compatibility

problems has resulted in a number of "home-grown" systems - most of which may be hardware or output format specific. Additionally, many of the available or "home-grown" capture programs do not offer the chain-node form of digitizing polygons.

The remaining part of this paper will discuss the advantages and disadvantages of using a CAD package for data capture, and the development of a post-processing program for relating attributes with objects and building of polygons from chains. The post-processing program described is also "home-grown" and may suffer from some of the deficiencies of other home-grown digitizing modules. However, the program is designed around a widely accepted and very portable CAD package. Further, one of the intentions of the post-processing program is to create the multitude of output formats demanded by the variety of cartographic and GIS packages.

Advantages of Disadvantages of CAD

CAD packages are currently used in cartography as an automated drafting package; thus, many cartographers are already familiar with at least one CAD package. Because of their present use as a drafting package, many academic departments or agencies already have the software. Also, CAD packages are relatively easy to learn. Most CAD packages have a myriad of display and editing commands that we cartographers wish existed in a data capture module. Many CAD packages support almost every available display card/monitor and digitizing tablet. Some CAD packages even have greater coordinate representation (i.e., number of digitis precision for a coordinate) than some of the most sophisticated and expensive cartographic data capture modules. With a CAD package, a map may be digitized in sections or as a whole; thus, resulting in a 'continuous' digital representation. Finally, many of the better CAD packages allow unlimited drawing complexity.

There are several disadvantages of using a CAD package for map digitizing. CAD packages generally do not allow chain-node digitizing of polygons. An identification attribute is not easily assigned to each point, line, or polygon - a process referred to as relating attributes to objects. CAD packages do not create an output format suitable for most mapping or GIS packages. Finally, some of the cheaper CAD systems do not allow true string or chain digitizing (referred to as a polyline in CAD) and have limitations on the complexity of the digitized drawing. Nonetheless, because of the relative availability, moderate expense, display and editing capabilities, and multitude of hardware configurations supported, a CAD package is an attractive platform on which to design a cartographic digitizing module.

AUTOCAD AS A CARTOGRAPHIC DATA CAPTURE PACKAGE

AutoCAD was chosen for this study as the CAD package to use for digitizing cartographic features because of the myriad of hardware configurations supported, the available capture and editing commands, and the stability of the product. The relative success of the company and product insures its support for years to come. Finally, many departments and agencies that require a cartographic data capture package already have a version of AutoCAD for other purposes.

Digitizing Logic

The digitizing logic of using AutoCAD and the developed post-processing package (named OADIG for Object and Attribute DIGitization) is shown in Figure 1. All objects and attributes on a map are digitized in AutoCAD and an output file created (a .DXF interchange file). The postprocessing package will read the AutoCAD output file and attempt to relate attributes to objects and assemble chains into polygons, with or without islands. If digitizing errors

are encountered, the OADIG program creates an AutoCAD file of error locations/types used as a graphic overlay onto the original digitized map. The geographic data is edited with AutoCAD and a new output file created for input to OADIG. The cycle of editing and building polygons/relating attributes continues until all errors are removed. When all errors are eliminated the OADIG program is used to create a suitable file in the format of many common GIS or cartographic packages.

*Set up layers to be used (e.g., POINT, LINE, AREA).
 Digitize all objects.
 Create .DXF output file.
 With OADIG, attempt to build polygons and link attributes.*

*Until OADIG indicates no errors,
 With AutoCAD,
 Input .DXF file of error layers onto original digitized map.
 Noting the types and locations of errors, edit objects in map.
 Erase objects in error layers.
 Create .DXF output file.
 With OADIG, attempt to build polygons and link attributes.*

With OADIG, select appropriate output format of cartographic objects.

Figure 1. The digitizing logic of using AutoCAD and the developed post-processing package (named OADIG for Object and Attribute DIGitization) presented in pseudocode.

The location of each point, line, and areal feature on a map is recorded using either the POINT or PLINE AutoCAD command. The POINT command is used to record entity points. The PLINE command is used to record strings for linear objects and chains for polygonal objects. The identification attribute of each object is recorded with the TEXT command as a label point. The point, linear, and areal features are recorded in three separate layers - appropriately name POINT, LINE, and AREA layers, respectively (Figure 2). The DXFIN and DXFOUT commands are used to create files for input to OADIG and to read OADIG created files into AutoCAD. For editing, the additional AutoCAD ERASE and PEDIT commands are used.

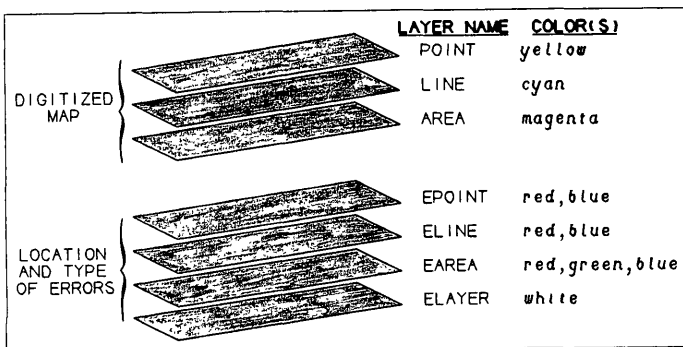


Figure 2. The appropriate layers for digitizing cartographic objects and label points and the error layers created by OADIG containing the location and type of errors encountered.

For an experienced AutoCAD user the commands identified above (with the exception of DXFOUT and DXFIN) should be familiar. A few simple digitizing rules using the described AutoCAD commands must be followed (Table 1).

Table 1
Criteria for Digitizing With AutoCAD and OADIG

Layer Recording:

- Objects and their attributes must be recorded on the appropriate layer: POINT, LINE, AREA.

Location Recording:

- Entity points must be digitized using the POINT command.
- Strings & chains must be digitized using the PLINE command.
- Polygons must be composed of one or more chains that together form a "ring" (i.e., they must not intersect and they must produce closure).
- The endpoints (i.e., nodes) of chains must perfectly join.

Attribute Recording:

- Label points of points, lines, and areas must be digitized using the TEXT command.
 - A label point of a point or line object must be closer to the point or line desired than any other point or line.
 - The label point of a complex area must be inside the outer ring but outside any inner ring.
 - Each inner ring of a complex polygon must also have a label point.
 - Each object must have one and only one label point.
-

Use of Layers

Layers are used in the described digitizing logic for a number of reasons. First, as all label points for the three types of features use the same command (i.e., TEXT), recording the labels on separate layers allows for differentiation (e.g., between the label point for a line and that for a polygon). Similarly, differentiation between strings and chains (both recorded as polylines) is required.

In general, most cartographic data capture packages do not explicitly use the concept of perfectly registered layers of point, line, and area objects. However, digitizing experience of the authors has demonstrated the usefulness of recording the location of certain combinations of geographic features simultaneously visible on the display screen - such as digitizing the location of a river that is also the boundary between two areas. Perfect matching of location for the river (string) and area (chain) is essential.

Finally, layers also allow for the convenient use of colors and separation of errors from original objects. As a convention, the three types of geographic features and the three types of errors are recorded in unique colors. The layers for points, lines, and areas are displayed in yellow, cyan, and magenta. As the error layer is drawn "on top" of the original digitized map last, the objects in error will be illuminated in the appropriate error colors. Red indicates an object without an attribute, blue indicates an attribute without an object, and green indicates a non-closing polygon. The OADIG program creates an error file with point errors displayed in the EPOINT layer, line

errors in the ELINE layer, etc. (Figure 2). Objects digitized in an incorrect layer are placed in the ELAYER layer. An entire layer of errors may be created and turned on (visible) or off (invisible) as necessary for interpreting the location/type of errors and for appropriate editing. When editing the objects, the appropriate error layers may be "turned off" allowing the original objects to be erased or altered. After the editing process is completed, all error layers are erased in AutoCAD and a new output file is created for another use of the OADIG software.

ALGORITHMS FOR RELATING ATTRIBUTES AND BUILDING POLYGONS

The algorithms for relating identification attributes (label points) to the appropriate objects and for the building of polygons from chains are discussed for the benefit of the user. If the user understands the algorithms, the errors are much easier to interpret and edit. This not only results in a better product, but a less frustrated user.

Relation of Attributes to Objects

The correct attribute (described by a label point) for each object is identified using a distance or polygon inclusion criteria. The correct attribute for a point is the nearest label point. Determination of distance from an entity point to a label point uses the simple Euclidean distance in two-dimensions. The correct attribute of a line is the nearest label point to any line segment in the string. A label point may be identified as within a given distance to a string using a path search around the line segments making up the line. A thorough discussion and implementation of the path search algorithm in the FORTRAN language may be found in Baxter (1976). A critical distance is used as a constraint to aid in relating label points and points or lines (Figures 3 and 4). This critical distance is necessary as either a label point or an object may be incorrectly omitted; thus, creating two different types of errors. The attribute for a polygon is the label point that is located inside the polygon. A complex polygon must have label points for the outer ring and for each inner ring (Figure 5). The point-in-polygon test for determining whether a label point is inside, outside, or on the boundary of a polygon uses the algorithm by Edwards and Coleman (1976).

Building of Chains into Polygons

The goal of the chain-to-polygon procedure is to create a set of polygons (either simple or complex) that are defined as a sequence of X-Y coordinate pairs representing line segments enclosing the area. If the polygon is complex (i.e., has inner rings) then the outer ring will appear first in the list, followed by the coordinate pair sequence representing each inner ring. This definition of a polygon is well suited to the requirements of most cartographic and GIS packages although may not be suitable to those GIS packages requiring DIME format chains. While the presented algorithm does not explicitly retain the left and right identifications of each chain, minor bookkeeping could be added to create a DIME format output.

Several algorithms exist for the building of string or chains into polygons (Cromley, 1984; Burrough, 1986). Cromley (1984) presents a method for digitizing conformant zones (i.e. polygons without islands) that requires the user to enter left and right areal codes as well as node identifiers. This additional user input is time-consuming although important for the data structure created. Cowen, et al. (1986) digitized polygon boundaries as chains and then used the GIRAS Arc-to-Polygon procedure to assemble the chains. The GIRAS interface worked well except for the mainframe requirements and

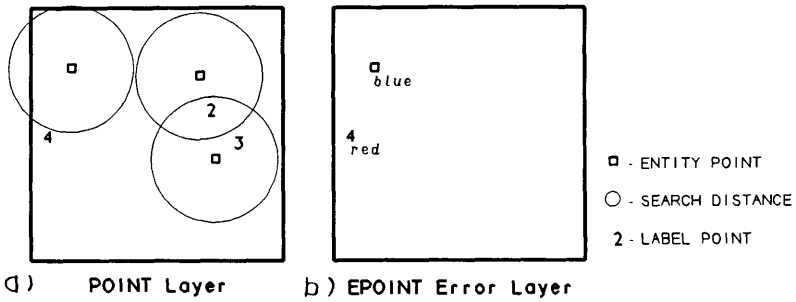


Figure 3. Determination of attributes to be associated with cartographic **point** objects. The original digital map is shown in a) and the resultant error layer produced by OADIG is shown in b). One entity point did not have a related label point and one label point (i.e., number 4) did not have a related entity point.

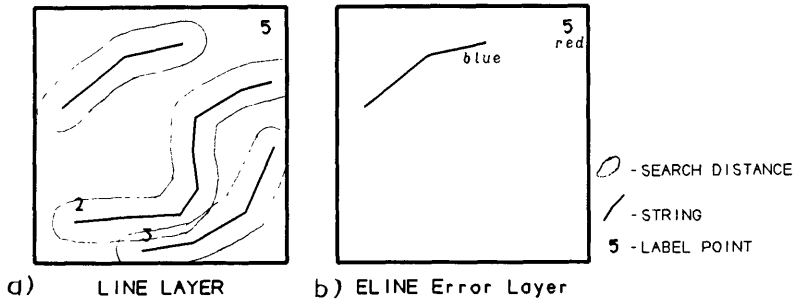


Figure 4. Determination of attributes to be associated with cartographic **linear** objects. The original digital map is shown in a) and the resultant error layer produced by ODIG is shown in b). The label point for a string was missing and the string for label point 5 was not recorded.

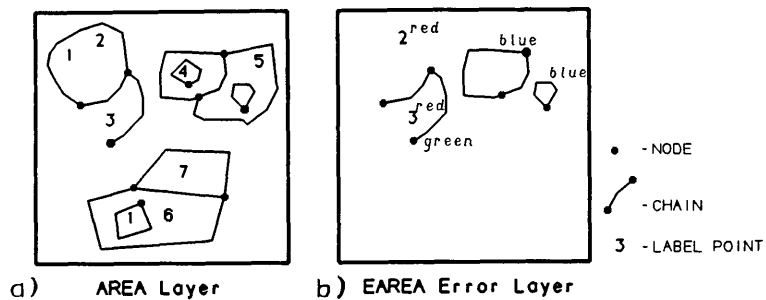


Figure 5. Determination of attributes to be associated with cartographic **areal** objects. The original digital map is shown in a) and the resultant error layer produced by ODIG is shown in b). The three types of errors that may occur when digitizing areal features are illustrated.

lack of interactive graphic editing. A somewhat abbreviated discussion of building polygons from chains and identifying the islands within each polygon is presented by Burrough (1986).

The algorithm developed and presented in this study began from a discussion with Fegeas (1985) on the GIRAS Arc-to-Polygon algorithm procedure for polygon building, although it is not identical. The presented algorithm assumes that the digitized linear boundaries are indeed chains - they do not intersect and begin and end at nodes. Thus, the user must insure non-intersection of a chain with itself or other chains and that the endpoints are nodes. Visual inspection easily solves the intersection constraint and the OSNAP function in AutoCAD may be used to force the chain to 'snap' to the node (i.e., critical point) of another chain. (If the chain does not begin and end at the endpoint of other chains, an error will be graphically identified with the OADIG program.)

The algorithm begins by retrieving one chain from the available set of chains (Figure 6a). This chain becomes the first chain in the current ring (Figure 6b). All other chains are examined to identify the chains that also joins at the ending node of the current ring. The chain with the smallest counterclockwise angle is identified and added to the current ring, reversing its direction if necessary (Figure 6b). The process of identifying the next chain ends when the ending node of the current ring is the same as the beginning node (Figure 6c). Each chain is used twice, once for building clockwise rings and once for counterclockwise rings (as in Figure 6d). After being used twice, the chain is discarded from the available set of chains (Figure 6f).

Only clockwise rings may be a true outer ring of a simple or complex polygon. Counterclockwise rings are only needed for describing the inner rings of a complex polygon. The clockwise constraint for a valid outer ring eliminates the counterclockwise rings built around adjacent areas (Figure 6e). This algorithm for building simple and complex polygons from rings is presented in pseudocode (Figure 7).

SUMMARY

The advantages and disadvantages of using a CAD package for digitizing cartographic data have been discussed in this paper. The logic of using the AutoCAD package for digitizing point, line, and area features as entity points, strings, and chains with the related attributes as label points was presented. A post-processing package for assembling chains into rings and then polygons and the relation of label points to objects was developed in the study and outlined.

ACKNOWLEDGEMENTS

The use of a CAD package for cartographic data capture in chain-node form was the original idea of David Cowen. The work presented here was based on exploratory research originally conducted by David Cowen, Lynn Santure, and the lead author of this paper at the Social and Behavioral Sciences Laboratory, University of South Carolina. The authors would also like to express their gratitude to Marty Garcia-Cotter of AutoDesk for his support in the implementation of the software and to Eva Nesmith of the Computer Sciences Department, University of Colorado for her suggestions.

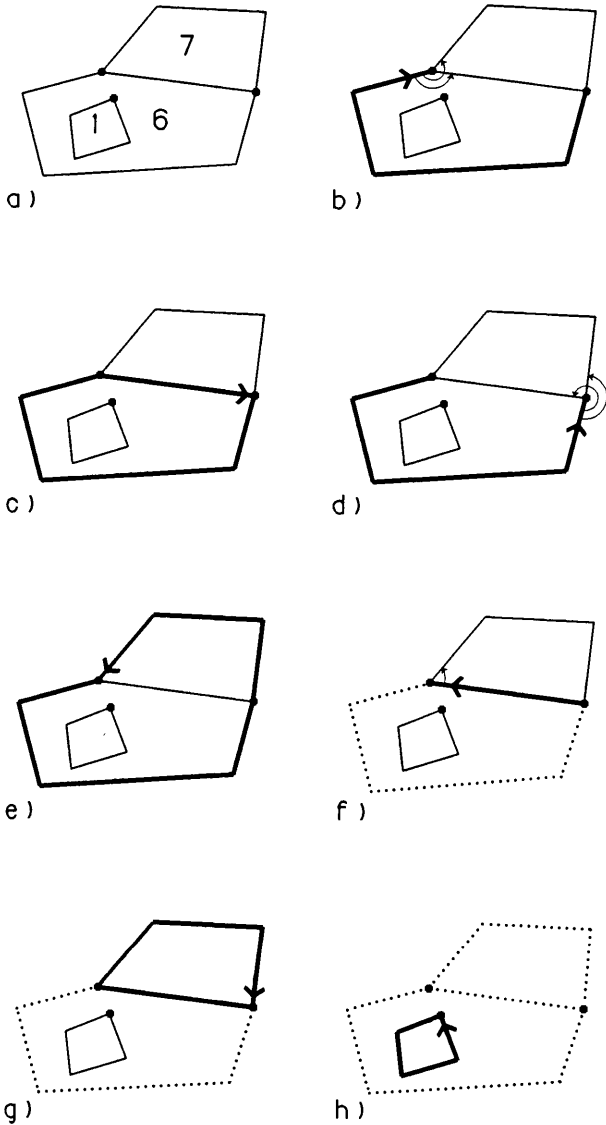


Figure 6. The sequence of steps for building a complete set of rings from chains. The original digitized chains are shown in a). The first chain encountered is used to begin building a ring in b). In c), the chain with the smallest counterclockwise angle is added to the current ring to complete a closed ring. Another chain is used to begin building the next ring (d) and angles are measured to identify the next chain. The second ring is completed in e). In f), the chain used twice has been eliminated and a new ring is begun. The third ring is completed in g) and the final ring is begun and completed in h).

For each ring,
 If ring is clockwise,
 /* This clockwise ring is referred to as the outer ring of a simple
 or complex polygon */
 Polygon = outer ring.

 /* Identify a candidate set of counterclockwise rings that are
 inside outer ring */
 Candidate Set = nil.
 For each ring,
 If ring is counterclockwise & inside outer ring,
 Add to candidate set.

 /* Add each appropriate inner ring in candidate set to polygon */
 For each ring in candidate set,
 If ring is not inside another ring of candidate set,
 Add ring to polygon.

Output completed polygon.

Figure 7. Algorithm in pseudocode for the creation of simple and complex polygons from rings identified previously.

REFERENCES

- Baxter, R.S., 1976. Computer and Statistical Techniques for Planners. (Methuen and Co., Ltd: London), 336 p.
- Burrough, P.A., 1986. Principles of Geographical Information Systems for Land Resources Assessment. (Clarendon Press: Oxford), 193 p.
- Cowen, D.J., M.E. Hodgson, L. Santure, and T. White, 1986. "Adding Topological Structure to PC-Based CAD Databases," Proceedings, Geographic Information Systems Workshop, Atlanta, GA, pp. 198-205.
- Cromley, R.G., 1984. "An Efficient Digitizing System for Encoding Conformant Zone Maps on a Vector Mode Device," Proceedings, International Symposium on Spatial Data Handling, Zurich, Switzerland, pp. 181-188.
- Edwards, P.G., and P.R. Coleman, 1976. IUCALC - A FORTRAN Subroutine for Calculating Polygon-Line Intersections, and Polygon-Polygon Intersections, Unions, and Relative Differences. (Oak Ridge National Laboratory: Oak Ridge, TN), ORNL/CSD/TM-12, 30p.
- Digital Cartographic Data Standards Task Force (DCDSTF), 1988. "The Proposed Standard for Digital Cartographic Data," The American Cartographer. 15(1): 1-142.
- Fegeas, R., 1985. Informal discussion of the GIRAS arc-to-polygon building algorithm, March 12.

TIGRIS MAPPER
VIEWED AS A DIGITAL DATA CAPTURING TOOL
IN OBJECT ORIENTED ENVIRONMENT

Jagdish Mitter
GIS, Intergraph Corporation
One Madison Industrial Park
Huntsville, Alabama, 35807
(205) 772-2000

BIOGRAPHICAL SKETCH

Jagdish Mitter was an officer in the Corps of Engineers of the Indian Army, where he received comprehensive training in land surveying and advanced photogrammetry in C.S.T.& M.P., Hyderabad, India. During his 18 years of experience, he commanded various units, conducting Topographical/Photogrammetric surveys in the most challenging terrains of India. After a distinguished career, he opted for premature retirement in the rank of Major in February, 1983.

He got his M.S. (Geodesy) in June, 1981 and M.S. (Math) with emphasis on Computer and Information Science from Ohio State University, Columbus, Ohio, (U.S.A.) in March 1985. He joined the Geographic Information Systems Development Division of the Intergraph Corporation in April 1985. Since then he has been actively involved in design/development of software packages for the InterMap Analytic (IMA) and TIGRIS.

ABSTRACT

An ideal Geographic Information System (GIS) looks for collection, storage, analysis and display of spatial and non-spatial data with speed, accuracy and consistency. To achieve this objective, Intergraph GIS has produced a number of packages. To convert the graphic information and digital images to a digital database, Intergraph has successfully developed a software package called Mapper. It is designed to take full advantage of the object-oriented programming techniques and platforms offered by Intergraph.

Data definition and organization is handled by Administrator. Three basic data categories that are used as the building blocks in the data definition are themes, composite features and base features. Base features are point, line, area and oriented line base features and have direct representation by virtue of their direct ownership of topology. The user can define attributes, attribute

type and default values associated with each theme, composite feature and base feature.

Mapper, being in the Topologically Integrated GeogRaphic Information System (TIGRIS) family of GIS system products, can capture raster and topologically structured vector data at data entry time with a little or no post processing. With capabilities such as extensive error checking/analysis for data integrity and topological consistency, queued edits and feature validation; Mapper can meet a variety of data requirements for many disciplines.

ADMINISTRATOR

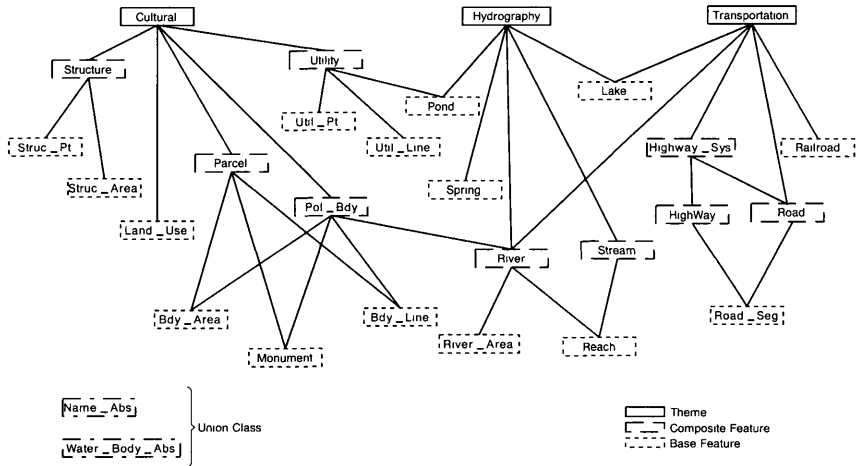
Spatial data in GIS occurs in points, lines and polygons or areas that are individual entities and serve as building blocks for all landscape features. Three basic categories in which data is defined are themes, composite features and base features. Data definition and organization of these categories is provided through Administrator, which serves essentially as the front-end database setup for Mapper. The objective is to provide the Project Manager with the capabilities needed to graphically define the class structure of the data and generate all the files necessary for Mapper initialization and start-up. This class structure can be converted to a paper menu for operators to use while they digitize. The graphical representation of the data and relationships between data is called a schema (Fig:-1). Since the user determines the structure of the data, he can create a schema that is unique to his cartographic needs. Three categories are briefly defined as:

Theme: A theme is a complete coverage of a single characteristic of the earth's surface. Themes can directly own any composite feature or base feature, and composite features/base features can be members of more than one theme.

Composite Feature: A composite feature is a representation of a geographic feature. Composite features can be 'components of' other composite features and/or themes as specified in the schema.

Base Feature: A base feature is a representation of a simple geographic entity. Base features are homogeneous in that no attribute of the base feature may vary within the extent of the base feature. Base features are point, line, area and oriented line features. An oriented line base feature is the same as a line base feature except that it has direction. Base features have direct representation by

Fig: 1 Example of an Administrator Schema



virtue of their direct ownership of topology. Base features can be owned by themes and/or by composite features. They are at the lowest level of the schema.

ADMINISTRATOR CAPABILITIES

IGE Primitives:

Placement/manipulation commands, windowing commands and edit commands.

Attributes:

Attributes are characteristics of themes, composite features and base features. The user can define attributes, attribute types and default values associated with each theme, composite feature and base feature. He can define attributes through code lists and edit attributes. If the schema has several elements that require the same attributes, the user can create an attribute grouping that has all the common attributes and union the attribute grouping with either composite features, base features or other attribute groupings.

Define Default Symbology:

To ensure the symbology is placed correctly and to save the operating time and the effort of defining symbology for each geometry, the project leader can define default symbology for themes, composite features or base features for subsequent use in data collection in Mapper. Point

symbols can be defined and stored in a cell library for use whenever desired.

Dictionary Creation:

From the graphical depiction of the data, a compiled dictionary is generated for use in Mapper data collection. All or a portion of the data may be compiled into a dictionary, and multiple dictionaries may be generated. For instance, one may wish to provide one master schema but support several data collection environments, say one for each theme, and therefore would graphically identify those portions of the data to be compiled into separate dictionaries.

Bulk Loading of Attributes:

This option allows the user to load the attributes into the schema from an ASCII file.

MAPPER

The strength of any GIS system depends upon the efficiency of its data capturing techniques, input/output subsystems, manipulation and analysis of data. Mapper is designed for manual conversion of hardcopy maps and digital images to a digital database, by utilizing a wide variety of graphics placement, manipulation and modification commands. The digital database comprises a cartographic feature structure with features that may be held in any combination of topological graphics. TIGRIS Mapper offers a set of facilities for geometric and attribute capture, as well as editing and manipulation. As the operator uses these commands to trace the cartographic features of a hardcopy map on a digitizing surface, feature data is recorded in the database; and images of the features are displayed on the workstation screen.

Mapper allows the user to choose two kinds of graphics that display identically but differ in the amount of sophistication of data recorded in the database. The first kind of graphics is IGE vector graphics, consisting of point, linear and area elements. For these elements the database records the x, y and z coordinates of input points and such simple symbology as color, line weight and line style. However, the database does not record attributes or retain any sense of element structure or spatial relationships.

The second and more intelligent kind of graphics is attributive, topological graphics created by the feature digitization capability. Each point, linear and area base feature so placed is defined in the database in terms of

its place within the feature and in terms of topological structure. The feature structure makes it possible to record attributes both at the base feature level and at a higher composite feature level. Topological structuring makes it possible to automatically record the spatial relationships among digitized features for the purpose of spatial analysis.

The user interface is simple icon-driven and reduces the need for the operator to have extensive training or system knowledge.

Mapper Platforms:

The workstation platforms on which the Mapper is built primarily consist of the following:

- o **Object Manager (OM):** For definition of object classes, instance data and the methods they possess;
- o **Query Subsystems:** For the definition of dynamic classes and instance data, and an SQL language interface to locate and extract information from these classes;
- o **The IGE and I/Draft Development Platforms:** IGE element placement/manipulation, windowing and edit commands.
- o **The Topological Platform (TOPO):** It defines a set of general classes for maps; themes; composite features; base features; relationships of point, line and area map features; and a set of feature classes to hold the geometric definition of these in a topological structure.

MAPPER CAPABILITIES

Coordinate Transformations:

Mapper is capable of supporting a variety of coordinate systems and provides the facilities to transform between them while maintaining topological integrity. Map projections, datum transformations and coordinate transformations provided in the Intergraph VAX-based World Mapping System (WMS) are supported by Mapper. It provides facilities for coordinate system definition, datum definition, single coordinate point readouts and feature placement with the coordinates of a secondary coordinate system.

Feature Placement:

This is a capability through which the operator can identify and manually digitize topological features already defined in the dictionary. Different digitizing modes, stream deltas and tolerances can be set before placing a feature. The user can store the commands in the system's memory in the order in which he wants to select; and by using the command stacking feature, he can suspend and resume multiple commands quickly without reselecting an icon.

Fig:- 2

The 'Place Feature' form is divided into several sections. At the top left, there is a title 'Place Feature' and two input fields: 'Keyin:' and 'Ownership:'. To the right of these fields is a grid of icons. The first row contains icons for 'Review Ownership', 'Shift', and a set of window control icons (minimize, maximize, close, and a checkmark). The second row contains icons for 'Change Status', 'Review Attribute', 'Query Char Attr', 'Locate Feature', and 'Paper Menu'. Below the icons, there are three main columns: 'Active', 'Classname', and 'Attribute'. Each column has a vertical input field. To the left of these columns are three buttons: 'Theme', 'Composite', and 'Base'. At the bottom of the form, there is a wide, empty input field.

The feature identification and placement process is carried out through the Place Feature Form (Fig:-2) and subforms. For operational convenience, options are provided for keying in class names, reviewing attributes of any feature class and placing a feature of the class already digitized. Change feature status and review attributes enable the user to have the appropriate class name or feature. He can query to find the attribute of a feature when the one displayed on the screen is not the one he wants or cycle through multiple features that have been found during a query for a characteristic attribute. The user can also select a theme, composite feature and/or base feature from a paper menu using the Paper Menu option.

While actually digitizing the feature, the user can borrow any digitized map feature or other graphic element. The borrowed feature becomes part of the geometric

definition of the feature he is digitizing as well as that of the original. This capability can be useful when part of a map feature, needed to be digitized, follows the geographical outline of another. The user can also append adjacent topological line segments to the feature being digitized.

Define Class Symbology: This option works through a form (Fig:-3) in the same way as define default symbology, explained in Administrator.

Fig:- 3

Define Symbology

Theme: transportation

Feature:

Foreground Color: [Paint palette icon] 0

Background Color: [Paint palette icon] 0

Line Weight: [Line style icon] 0

Line Style: [Line style icon] 0

Fill Pattern: [Square icon] 0 [Dropdown arrow]

Point Symbol: [Square icon] [Dropdown arrow]

Composite: [Empty preview window]

select button for subform or field for keyin

Registration Setup:

Maps are two dimensional representations of topography; therefore, there exists a direct translation of all topographical entities into the equivalent elements of planar geometry (i.e. points, lines or polygons). Hence, they form one of the major sources of information for any GIS database. Mapper allows the user to define a coordinate transformation between a hardcopy or a raster image and the mapping plane. The user can input control points and check points and their standard deviations in the database, digitize each control point several times and perform a four- or six-parameter transformation to map the digitizer surface to the design file coordinate system. Mapper supports an inquiry process so that the residuals and the parameters of the transformation may be studied and

iterated until the transformation is rejected or accepted. The user can edit the set up, control points and residuals lists. He also has the option to select the cursor shape and set/reset track areas.

Feature Edits:

A set of commands exists for the manipulation of data within the TIGRIS design file. These commands provide the workstation operator full control to enter, modify and maintain element and attribute data. Since TIGRIS correctly maintains all topological relationships automatically, the operator is free to concentrate on the correctness of the geometric and attribute data.

Attribute Manipulation and Ownership Options:

These capabilities apart from saving user time, enhance flexibility and operational efficiency. While editing features belonging to the same class, the operator can compare, review and copy the attributes of one feature to another. The change owner option enables him to attach a component to additional owners or detach from the current owner. With the connect to owner option the user can connect a base feature to an additional owner, whereas the connect to component capability allows him to connect a composite owner feature to an additional feature. A component can be a base feature owned by a composite feature in the feature structure or a composite feature, higher in the hierarchy.

Display and Reporting:

Mapper provides a variety of display and reporting capabilities to enhance the representation of TIGRIS data. The user can display, highlight or fill the geometry of themes, composite features, base features or the results of queries. All display symbology is user-defined. TIGRIS provides for the definition of display hierarchy rules that are applied when different features share topology. This permits the user to choose which classifications of information take precedence over others in the display. TIGRIS reporting facilities allow the user to define report formats for features selected by attribute or query. Users provide a representation of the content and appearance of a report through interactive methods prior to its generation.

Feature Validation:

To ensure quality control and digitizing standards, Mapper provides capabilities for validation of geometric and attribute data. Anomalies detected can be posted to a queue for subsequent review in the Queued Edit (QE) subsystem.

Attribute Validation: Mapper provides for verification of attribute data to the user through the build query, select set and select query options. Build query enables the user to build a query that searches for all the members of a class based on any combination of attribute values (composite or base feature). The data found like this may be placed on a queue, saved in a result set or input to a graphics command such as move or delete. The query itself may be saved for re-execution later. All or portions of the dataset may be queried. The select set and select query options provide the appropriate functions for reviewing sets or queries, copying sets to queues, outputting sets to graphics or re-executing queries. In addition, an interface to Structured Query Language (SQL) is provided through the SQL editor, which provides the operator with the capability to perform the standard SQL functions on the data such as update, select, etc.

Geometric Validation: This option locates digitization anomalies (errors) such as node mismatches, undershoots, overshoots and slivers. The user may invoke any combination of these anomaly detectors by specifying a UOR tolerance for each selection. The process may be run against the entire dataset or some portion defined by the user. Digitization errors found using the geometric validation are automatically sent to a queue called geometric anomalies. To review/edit the errors in the queue, the QE commands are used.

Queued Edit (QE): QE provides a means for viewing and manipulating a queue of data items. An item is a location in the design file that has been recorded in a queue, built by the attribute or geometric validation, according to a criteria defined by the user. The QE command displays the locations one by one for the user to view, and if necessary, edit with standard topological graphics placement and manipulation commands and make necessary changes in associated attribute values. QE also allows the operator to interactively create queues and subqueues, and define items to be stored in them.

Translate In and Translate Out:

Capabilities exist to translate IGDS/Relational Data Base (RDB) design files to TIGRIS design files or TIGRIS design files into IGDS/RDB design files. A capability exists to take IGDS/RDB into and out of IGDS/DMRS. TIGRIS design files can be generated from Neutral File Format (NFF) and vice versa (i.e. NFF to TIGRIS design file). Separate

translators are also available to support input or output to and from formats such as USGS Digital Line Graph (DLG) and Initial Graphics Exchange Specification (IGES).

CONCLUSION

It is evident that with Administrator the user can define and organize his data according to his requirements, and Mapper provides the capability to capture data with speed, accuracy and consistency. Object-oriented programming philosophy in the development of GIS software packages provides tremendous flexibility and capability to meet the future challenges of the dynamically changing industry.

ACKNOWLEDGEMENTS

The author wishes to thank Richard Bevis, Peter Ring, David Glenn for their contributions in writing this paper and Debra Clifford for her valuable suggestions.

DATA CAPTURE FOR THE NINETIES : VTRAK

Robin Waters, Dr Dave Meader, Greg Reinecke
Laser-Scan
12343F Sunrise Valley Drive, Reston, VA, 22091

ABSTRACT

Geographic Information Systems (GIS) need cost effective data capture systems to feed their voracious appetites for data. Modern GIS require both raster (grid) data and vector data. Raster scanning of existing maps is a very attractive means of capturing data but the conversion of such images into intelligent, attribute-coded and topologically structured vector information is a non trivial exercise that has confounded many GIS project leaders. Laser-Scan pioneered interactive vector scanning with FASTRAK and LASERTRAK systems and have now implemented the same pragmatic and cost effective algorithms and techniques on a standard workstation accepting input from a variety of scanners. VTRAK, now in use in N. America, Europe, and Japan, is a unique system for applying the best algorithms to the right set of data at the right time. Fully automatic "vectorisation" or "raster to vector" programs must take a global approach to the raw data whereas VTRAK applies image processing, feature extraction, line following, symbol recognition and other techniques in a selective approach. All these can be applied at a standard workstation in different sequences depending on the source documents and the target database. This paper describes VTRAK, its justification, philosophy, operation, application and future development.

JUSTIFICATION

Map data capture is big business. In North America the map data conversion business will be worth more than a billion dollars over the next decade.

Utilities are investing in very large databases for geographic referencing of outside plant records.

Telephone companies are not only improving their records but automating the optimisation of their taxes.

Defense agencies are converting map products into digital form for faster updating, ease of use and direct input to some weapons systems.¹

Federal government, in the guise of many different agencies needs topographic and thematic information in GIS to provide a more cost effective service for the taxpayer.

Commercial initiatives range from the direct use by forest companies to the large number of data conversion services now available.

Despite increasing amounts of data becoming available in digital form direct from original surveys (satellite imagery, digital photogrammetry & total station surveys), the need to convert existing maps continues to grow.

UNIQUENESS OF MAPS

Maps have certain characteristics that distinguish them from, for example, engineering drawings (Woodsford 1988). These characteristics, detailed below, have a decisive effect on the methodology for converting them into digital form.

Maps are **accurate and to scale**. Measurement from the film or paper is the best that can be obtained; there are no dimensions written on the map. However, correction of data to a pre-existing grid or graticule will be vital.

Maps contain **very fine detail**. Lines of 0.1mm (4mil) or finer are commonplace. A high resolution scanner and the necessity for handling large images are required.

The **variety** of symbols and linestyles within a map series is very great. Worldwide the variations are almost infinite.

Maps are often **multicolored** with no access to the source separations.

Maps are generally part of a **continuum** with abutting or overlapping neighbours. Edge matching is non-trivial.

Maps are **multi-purpose** documents and conversion must be carried out with this in mind. Most individual users requirements are less expensive subsets of a general purpose conversion exercise and careful attention to specification and quality control is needed to ensure the support of a range of users.

Maps are of **variable quality**. Best results for data capture will always be obtained from scribed film originals on stable base material. However only blueline paper copies may be available, often overwritten with confusing detail.

Laser-Scan have been dealing with maps for over 15 years and recognise the unique nature of maps and the requirements of the users of geographic information whether it is to be used for the reproduction of maps by digital methods or for recording and analysis of data to produce answers to many geographic queries.

USER REQUIREMENTS

Vector data for digital mapping and GIS can vary in complexity from simple "spaghetti" linework to multi-attribute, object level, topologically structured data. The data capture process is not complete until the product has been tested and proven to conform to the specification and checked against the original input document.

The stages in a data capture process at which increasing complexity is added and/or at which quality is checked are critical to the efficiency of the process. Failure to check basic geometric accuracy until complete topological structuring has taken place will lead to massive reprocessing. However introduction of basic topology at

the time of vector feature extraction can help to achieve correct geometry at the first attempt.

VTRAK procedures can be designed to help this overall process and in particular to deal with the "hidden agenda" behind the overt specification (Woodsford 1988).

Accuracy. Typically specified as "half a line width" or perhaps "2 mil", accuracy normally includes implicit criteria for shape and "cartographic acceptability". VTRAK algorithms and immediate interactive overlay checking ensure quality.

Representation. Different features require different digital representation that can only be ensured by recognition of the class of feature prior to extraction from the image. VTRAK provides this facility. In fact, as a result, completely different algorithms and parameters can be applied to different features.

Abstraction. Many GIS objects are inferred from the actual map depiction; they are not explicit. The centre line of cased roads, the centre point of a symbol, and the indefinite bounding polygon of a symbolised swamp are all particular examples handled by VTRAK.

Selection and Completeness. Different data capture specifications applied to the same map will require that a selection process is carried out. Failure to do this at an early stage in conversion will lead to wasted and often counter productive processing. VTRAK blends selection of parts of the image for background processing with overt interactive selection during feature extraction and special vector edit functions to optimise selection.

Quality Control. The cost of conversion varies in direct relation to the quality control procedures used. The quality assurance programme carried out by the end user or his agent is a factor that must be taken into account in any conversion programme. A VTRAK system can be tuned to handle the most stringent tests including random remeasurement, checking of feature quality with respect to attributes and topology checking.

VTRAK OPERATION

Configuration. VTRAK operates on standard Digital Equipment Corporation (DEC) VAXstations but with background processing, scanner interfacing and network handling capable of being carried out on other VAX processors in a cluster. Laser-Scan is a multi-national OEM for DEC and can supply complete turnkey systems anywhere in the world. The one additional item added to the workstation for VTRAK operation is the Mapstation Console. This replaces the mouse as a pointing device and provides extra function buttons which are the most ergonomic means of interacting with the display. All modes of VTRAK provide pop-up menus; keyboard input is only needed for the entry of file names.

Scanning. VTRAK will accept binary or greyscale raster data from a variety of scanners which may be chosen for speed, accuracy, resolution and other parameters to handle the required maps. Transfer to the immediate VTRAK environment can be by tape, network or direct on-line connection. Laser-Scan will configure particular scanners for turnkey systems. Parameters that need to be known for each image are type of scanner, image polarity, scanner resolution and size (in pixels) of the image. Small parts of the image can then be viewed to set parameters for thresholding, orientation, color separation and zoom factors.

Image Zone Definition. Some whole images and some areas of other images are suitable for fully automatic 'Autopass' feature-extraction in background mode. The areas to be selected for this are zoned by coloring them green at this stage with a variable sized rectangular window interactively moved across the image.

Autopass. VTRAK background mode feature-extraction is called 'Autopass'. While most of the feature extraction parameters are identical to those used in an interactive mode, there are additional functions such as differentiation of lines by their width, selection by minimum size and creation of an extra file of edit requests, where the automatic process realises that human intelligence will be required.

Interactive Feature Extraction. The heart of VTRAK is the interactive mode used for setting all feature extraction parameters and for interactive (but semi automatic) extraction of features selected by the operator. This mode is used for all areas not zoned green for autopass. In fact autopass can be delayed until after an interactive session so that difficult features can be extracted before the background process 'screws them up' ! Interactive VTRAK typically requires the following simple operations :

select feature with cursor controlled by tracker ball

press precoded start button that selects suitable parameters for that type of feature, codes the feature with the appropriate attribute(s), and starts the feature extraction process.

watch process of feature extraction which can be halted or reversed at any time and which can be displayed at any zoom factor.

guide process (if desired) in complex areas. At this stage VTRAK can also accept features digitised 'manually' off the screen image using the cursor at any zoom factor. This is how 'imaginary' features may be input or very low quality image data handled.

accept feature when completed. A single button press signifies the end of that feature, the writing of that feature to the output vector file and the 'paint-out' of that feature from the raster image so that it will

not be repeated or be able to confuse further processes. (Paint-out also happens in the Autopass mode).

edit features with image background. Presentation of the complete set of features overlaid on the original image gives a final edit and check capability with several special purpose functions. Exact registration is achieved via corner points or other registration marks on the image and a systematic pan and zoom routine gives a completeness and accuracy check. Procedures for checking attributes by selective display, for showing topological queries and for sensible handling of text and symbol features can also be introduced. Automated attribute coding for multiple-coded features or the building of 'objects' from basic features can also be supported.

output processing. VTRAK is a data capture system that can provide inputs to a variety of GIS or digital mapping systems including Laser-Scan's own comprehensive suite of programs. Other systems supported are Arc-Info, Intergraph, Synercom, Autocad and general purpose output to USGS DLG files or a variety of military formats.

VTRAK feature extraction algorithms are proprietary and do not use the 'skeletonising' approach used by many raster to vector systems. They use all available pixel information to extract the centreline of a multi pixel feature or the centre of a symbol. Similarly, nodes in line networks are extracted from an array of pixels and not just from the junction of skeletonised lines. VTRAK does not suffer from 'hairs' and 'junction kinks' often seen in standard vectorised output. VTRAK algorithms are also designed to 'fail safe' and call for help rather than struggle with bad data.

The emphasis of the VTRAK philosophy is toward flexibility stemming from the recognition that no two map specifications are alike and that every map is unique in itself. VTRAK provides managers with the ability to give the actual operators of the system as much or as little flexibility as their training and the type of work demand. Tuning of parameters can be reserved for management or left to the experienced operator.

NORTH AMERICAN APPLICATIONS

VTRAK has been benchmarked in N. America against competing products for utility and topographical mapping applications. These benchmarks have shown that VTRAK is superior to all other products as a cost effective scanned data capture system. VTRAK is now installed at the Canada Centre for Geomatics (Energy Mines and Resources) where it is used primarily for the 1:500000 series.

USGS 'quad' sheets are ideal sources and some special features of VTRAK are applicable to these maps :

single operation removal of contour labels and joining of contour gap across label.

simultaneous heighting of multiple contours.

recognition, orientation and coding of multiple house symbols.

measurement of road centre lines from cased road representation.

extraction of all types of dashed & dotted lines (eg boundaries, tracks, intermittent rivers).

topological structuring of river and road networks to DLG standards.

Some of the facilities are used on the LASERTRAK scanners at USGS where contours (hypsoigraphy) and rivers (hydrography) are being captured for the National Cartographic Database (Moreland 1986).

Utility maps require a different emphasis because they typically have less linework, more text and symbols and are often of lower quality. There are no simple answers but VTRAK gives a superb environment for the development of very efficient routines with menu, keyboard or voice entry of attributes, on-line intervention in critical areas and creation of network topologies. Both land base and plant data can be captured and kept separate so that, for example, the land base can be sold to other utilities, local government and other users (Cross 1987).

Thematic maps of soils, geology, water resources, geodemographic data, vegetation, city zoning boundaries etc. are vital inputs to GIS. VTRAK not only enables the user to input the linework with basic coding, but also provides polygon building and coding facilities.

EUROPEAN APPLICATIONS

Most of the facilities described above apply equally to European maps except that the range of map specifications is much greater and perhaps more centrally regulated. Laser-Scan have been very involved in setting the UK standards for digital mapping and have developed specialised routines for digitising Ordnance Survey large scale maps (1:1250, 1:12500) and for quality controlling the process to the National Joint Utilities Group (NJUG) standards.

In continental Europe the base mapping for local government and utility applications is the cadastral map used primarily for recording land ownership and for taxation. These maps are highly accurate and typically feature parcels surrounded by boundary lines which connect beacons denoted by hollow circles. The accurate

measurement of these circles, their topological connection to others via the boundary network and the addition of attributes to both beacons and parcels are fundamental to the data capture process. VTRAK has 'beacon recognition' facilities and enables the operator to capture a parcel at a time together with associated attributes.

CONCLUSION

During the 1970s extravagant claims were made about the efficacy of raster scanning and vectorisation : Laser-Scan developed the FASTRAK line following scanners, which are still in use today.

During the 1980s realisation that current raster scan capabilities were way ahead of the software for raster to vector conversion led to the increased use of raster mode drawing management systems and to the proliferation of conversion houses using well tried but time consuming manual digitising systems. Laser-Scan introduced the LASERTRAK vector scanner which is in use in the USA, UK, Japan, Italy, Norway and the Middle East.

During the 1990s both raster drawing management and vector database systems will coexist and will need to exchange data for efficient use of GIS resources. Laser-Scan have introduced VTRAK to meet the need for feature extraction from raster map images as and when it is required.

VTRAK is available now !

REFERENCES

- Cross D.A. 1987, Intelligent Scanning of Maps and Plant : European AM/FM Conference, Montreux
- Moreland D.K. 1986, Development of an Automated Laser-Based Data Capture System at the US Geological Survey : ACSM-ASPRS Spring Convention, Washington
- Woodsford, P.A. 1988, Vector Scanning of Maps: Proceedings of IGC Scanning Conference, Amsterdam

POLYGON OVERLAY TO SUPPORT POINT SAMPLE MAPPING: THE NATIONAL RESOURCES INVENTORY

Denis White
NSI Technical Services Corporation
200 SW 35th St.
Corvallis, Oregon 97333

Margaret Maizel
American Farmland Trust
1920 N Street, Suite 400, NW
Washington, DC 20036

Kelly Chan, Jonathan Corson-Rikert
Laboratory for Computer Graphics and Spatial Analysis
Graduate School of Design, Harvard University
48 Quincy St.
Cambridge, Massachusetts 02138

Problem

The construction of timely and pertinent policy for wise use and preservation of agricultural resources is predicated on an adequate knowledge of the status and extent of these resources. For nationwide policy development, it is therefore important to have comprehensive national surveys of natural resource information using uniform criteria. Such surveys of non-federal lands of the United States have been conducted by the Soil Conservation Service (SCS) at five different times in the last 30 years. The most recent of these was the 1982 National Resources Inventory (NRI).

The 1982 inventory consists of approximately 800,000 sample points in a statistical sampling design, each recording over 100 variables indicating soil, agricultural, and land use characteristics (Goebel and Dorsch, 1986). Each sample point is geographically referenced, in the distributed version of the survey, by the county, SCS Major Land Resource Area (MLRA), and US Geological Survey (USGS) hydrological cataloging unit in which it lies. (Latitude, Longitude coordinates for sample points are recorded by SCS.)

The statistical design of the inventory is such that the relative error associated with an estimate of area for a given crop or land use reduces as the area of the reporting unit (and hence the sample size) increases. So estimates for states have larger error bounds than those

for the entire country, and MLRA's and counties have successively larger bounds yet. Partly because of this characteristic, and partly because the design was optimized to report at the scale of MLRA's, little analysis of the 1982 NRI has been performed at scales of resolution finer than MLRA's.

Nevertheless, valuable information is contained in the NRI for even sub-county scales of resolution if interpretations are restricted to statements about the sample points, or aggregates of them. This paper describes a process for obtaining useful interpretations through geographic analysis techniques using geographic information system technology.

Method

One approach to NRI mapping is with raster methods. The georeferencing coverages (MLRA's, hydrological units, and counties) are rasterized separately into three index layers. An index layer contains pointers to (or geocodes for) individual MLRA's, hydrological units, or counties. NRI sample points are then grouped by the triplets of units of the three coverages that actually appear in corresponding cells of the rasterizations. Appropriate aggregations are then applied to NRI variables for the grouped sample points and the aggregated values are assigned to a raster layer.

We opted for the precision and flexibility of the analogous vector approach, that is, polygon overlay. In this case least common geographical units (lcgu's, Poiker and Chrisman, 1975) are created from the intersections of the MLRA, county, and hydrological unit coverages. Each lcgu is a polygon contained in a single MLRA, county, and hydrological unit. As with the raster approach, aggregated variables for sample points contained in each unique combination of the three coverages can then be mapped.

Federal lands can be excluded in the raster approach by rasterizing a binary coverage of federal/non-federal and excluding federal cells from NRI mapping. In the vector approach, the federal coverage can be overlaid upon the other three to obtain a similar exclusion.

Databases

Major Land Resource Areas (USDA SCS, 1981) are geographic regions that have similarity in natural resource characteristics as applied to agriculture, forestry, engineering, recreation, and other land uses. The characteristics used to define MLRA's are land use,

elevation and topography, climate, water resources, soils, and potential natural vegetation. There are 204 of these regions ranging in size from about 2,000 km² in the California Central Valley Delta to about 280,000 km² in the Northern Rocky Mountains (Figure 1). (For other multi-factor natural resource regionalizations see Bailey, 1976, and Omernik, 1987). The digital coverage of MLRA's was obtained from the SCS office in Fort Worth, Texas in DLG format. The published map of MLRA's is at a scale of 1:7,500,000.

Major Land Resource Areas

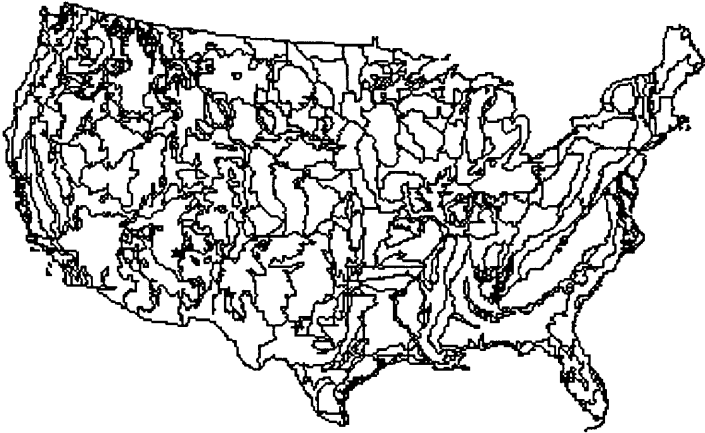


Figure 1

USGS Hydrological units are areal units that aggregate watersheds into areas of hydrographic similarity for research and resource management purposes. Cataloging units are the most detailed of a four level spatial hierarchy of units, totaling about 2100 polygons for the conterminous US (Figure 2). The digital coverage for these units was digitized from the two sheet USGS map of them at a scale of 1:2,500,000.

USGS Hydrological Cataloging Units



Figure 2

Federal land boundaries were obtained from the 1:2,000,000 scale DLG database distributed by USGS. A coverage for the conterminous US was assembled from the 15 sectional files in which this database is distributed by edge-matching along section boundaries. For purposes of this project, the various federal land ownership categories were aggregated to create a binary federal/non-federal coverage.

The coverage for counties of the conterminous US was a version of the Bureau of Public Roads county file digitized in the 1960's and distributed through the Bureau of the Census as the DIMECO file. This database had been maintained at the Lab for Computer Graphics for many years. The maximum suggested scale of use is 1:1,000,000 (Edson, 1984). The 1:2,000,000 scale DLG's were not used for this coverage because certain coastal county boundaries in this version follow legal definitions and lie in adjacent coastal waters, because edge-matching across sections would have been required, and because the resolution of the existing DIMECO file was adequate for this project.

All four coverages were (re)projected as necessary to the USGS standard Albers projection for the conterminous states and then coarsened (i.e., geometrically aggregated, Morehouse and Broekhuysen, 1982) to a resolution of 1.5 km. The resolution criterion derived from a consideration of the source scales of the coverages and expected final mapping scale.

Process

The polygon overlay of the four coverages (taken two at a time) resulted in a coverage of some 60,000 polygons using an overlay tolerance of 1 km. This file was coarsened to about 35,000 polygons using a tolerance sufficient to eliminate polygons smaller than about 1.5 km².

The overlaid geometric coverage was accompanied by a cross reference file relating the lcpu's to the containment units in the original coverages. This file was the basis for aggregating NRI data into appropriate attributes for mapping. The NRI sample point data file was sorted by its three georeferencing keys (MLRA's, hydrological units, and counties). The polygon cross reference file was sorted by these three and by the federal/non-federal land key.

The assignment of one or more mapping values to overlaid polygons consisted, conceptually, of synchronized passage through the two sorted files, aggregating attribute values for all sample points with the same unique combination of the three keys and assigning these values to all polygons cross referenced to the same unique combination (and in non-federal land). The actual implementation used an indexed sequential file for the polygon cross references to optimize performance.

A critical part of the process was the method of aggregation. Each NRI sample point includes an attribute called the "expansion factor" that records the number of acres the sample point represents. It is this factor, derived from the sampling process, upon which area estimates for land use and crop categories are based. In aggregating attributes, the expansion factor was used to calculate weighted averages of percent land in the various categories of the attributes represented by the sample points.

For example, one of the NRI attributes indicates whether the sample point is on land that meets prime farmland criteria. The calculated variable for overlay polygons is the percentage (ratio) of sampled land that meets prime farmland criteria and is computed as the ratio of the sum of each point in prime farmland times its expansion factor to the sum of all points, each weighted by its expansion factor. The title of this variable should then be something like "percent sampled land that meets prime farmland criteria".

Extrapolation of these percentages to produce an actual area estimate for an entire overlay polygon would be accompanied by relatively large confidence limits because of the relatively small sample sizes.

Since a choropleth mapping technique tends to imply uniformity of a statistic for a polygon, the qualifying titles for the maps are important.

Results

The Conservation Title of the Food and Security Act of 1985 (The 1985 Farm Bill), in instituting the Conservation Reserve Program (CRP), set up an important new mechanism to help conserve highly erodible or marginal cropland by restructuring price support payments for non-production on such lands into payments for placement of the same land into a conservation reserve for a ten year period.

The determination of eligible land for the CRP consists of a complex formula involving attributes sampled by the NRI. There are three criteria, the satisfaction of any one of which confers eligibility (7 CFR Part 704, Federal Register 2-11-87): cropland with an erodibility index greater than or equal to 8; or cropland in soils capability classes II through V with soil loss tolerance factor greater than 3T (three tons per acre per year tolerance); or cropland in soils capability classes VI through VIII.

The NRI provides a way to monitor the performance of the CRP, and the polygon overlay of the georeferenced databases allows a fine scale cartographic depiction of eligibility and performance (seen in coarse scale in Figure 3).

Percent Sampled Cropland Eligible for the Conservation Reserve Program



Figure 3

References

- Bailey, R.G. 1976. Ecoregions of the United States. USDA Forest Service. Ogden, Utah.
- Edson, D.T. 1984. Data Bases. U.S. National Report to the ICA, 1984. Special Issue of *The American Cartographer*.
- Goebel, J.J., Dorsch, R.K. 1986. National Resources Inventory: A Guide for Users of 1982 NRI Data Files. USDA SCS.
- Morehouse, S., Broekhuysen, M. 1982. *Odyssey User's Manual*. Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.
- Omernik, J.M. 1987. Ecoregions of the Conterminous United States. *Annals, Association of American Geographers*, 77:118-125, and map supplement.
- Poiker, T.K., Chrisman, N.R. 1975. Cartographic Data Structures. *The American Cartographer*, 2(1):55-69.
- USDA SCS. 1981. Land Resource Regions and Major Land Resource Areas of The United States. *Agriculture Handbook* 296.

HAZARDOUS WASTE DISPOSAL SITE SELECTION
USING INTERACTIVE GIS TECHNOLOGY

Calvin Van Zee
Ebasco Services, Inc.
10900 NE 8th St., Bellevue WA 98004

John E. Lee
QC Data Collectors, Inc.
777 Grant St., #111, Denver, CO 80203

ABSTRACT

Site selection for disposal of hazardous waste requires consideration of numerous geographic factors. Use of a Geographic Information System (GIS) can facilitate examination of interaction between site-related factors. The IBM PC AT-based GIS STRINGS(tm) was used to cartographically model and identify potential landfill disposal sites for hazardous waste on the US Army's Rocky Mountain Arsenal in east-central Colorado. A digital database was created from multi-source maps of topography, hydrography, geology, transportation, utilities, land use/status, and man-made features. Several data themes were derived using reclassification, compositing, proximity, and polygon overlay analysis techniques. A cartographic model was built to identify potential sites based upon user-defined criteria. Interactive computer sessions with technical experts were used to further refine the model and test site alternatives. The "slide" function of the GIS allowed rapid interactive viewing of site-related factors. Executive decision makers were able to identify and choose between site alternatives in 3 1-hour computer sessions that might otherwise have taken months using manual cartographic methods.

INTRODUCTION

Disposal of hazardous waste has become a pressing environmental concern. Selection of a disposal site requires consideration of numerous geographic features and their interaction, and usually involves numerous governmental agencies with varied siting criteria. U.S. Army scientists at the Rocky Mountain Arsenal (RMA) near Denver, Colorado studied the feasibility of different hazardous waste disposal options. One possibility which was considered was on-site landfill. Factors which were considered in selecting a landfill site for disposal of these wastes included topography, transportation, utilities, land status, hydrography, geohydrology, and man-made features. Consideration of all of these factors was an enormous task, even on the relatively small area of the RMA (17,000 acres).

Use of a (GIS) permitted editing, analysis, and display of more site selection criteria combinations than would have been possible with hand-drafted methods. In addition, interactive computer sessions with the GIS database allowed executive decision makers to quickly modify criteria and identify site alternatives on a real-time basis. Producing these site alternatives with traditional engineering drafting methods would have taken months of manpower effort.

METHODS

Hardware/Software

The STRINGS(tm) GIS was used for this project. STRINGS (GeoBased Systems, Inc., Research Triangle Park, NC) is a topologically-structured (arc/node) vector-based GIS package with digital data capture (map registration and digitizing), attributing (primary and multiple), element editing (interactive and batch), database management (edge-matching and rubber-sheeting), polygon formation (with island identification), display (interactive and hard-copy), reporting (query and hard-copy), and cartographic analysis (reclassification, overlay, and distance) capabilities.

STRINGS operates on IBM AT (or compatible) personal computers under MS-DOS. This study used a Sperry IT CPU with 640 KB RAM, a 30 MB hard disk drive and 1.2 MB floppy disk drive, an 80287 math co-processor, serial/parallel ports, a monochrome text monitor, a high-resolution (1024 x 1024) color (RGB) 19" graphics monitor with Vectrix Pepe graphics controller, and a MicroSoft Mouse.

Map digitizing was accomplished using a high-resolution (0.005") large-format (48 x 60") Calcomp 9100 digitizing tablet with power stand, backlighting, and magnifying (5x) 16-button cursor. Hard-copy plotting was accomplished using a high-resolution E-size Calcomp 1073 4-pen plotter.

Database Creation

Several source maps were used in preparation of the geographic database for the RMA. These maps were at various scales, on different media, and containing various types and amounts of information. The first step in database creation was to identify the data themes to be captured, as follows:

- Topography (elevation contours at 5' intervals)
- Public Land Survey (township, range, and sections)
- Ground Water (ground water contours at 10' intervals)
- Bedrock (bedrock contours at 10' intervals)
- Land Status (exclusions due to ownership/development)
- Hydrography (lakes/basins, creeks, and ditches)

- Flood Plains (100-year flood plain outline)
- Transportation (paved and gravel roads, and railroads)
- Electricity (main and distribution electrical lines)
- Pipelines (oil and gas pipelines)

Data for each of these themes was captured separately and stored in a separate map file. Only data within the boundary of the RMA was captured, thus the boundary acted as a common border for all map themes. A primary integer attribute code was also assigned to each feature within a theme.

All data was captured in Colorado State Plane Coordinates (SPC) due to their suitability for engineering applications and anticipation of new surveying data being incorporated in the digital database. Maps were registered to the digitizing tablet using up to 16 points with known State Plane coordinates. Where no known points were available, USGS 7.5' quadrangle maps were registered and used to determine State Plane coordinates for points in common between the USGS map and the source map. These points were often section corners, road intersections, or stream confluences.

After the maps were registered to the digitizing table, an operator digitized the data as either lines or polygons. For example, streams were captured as lines and the flood plain was captured as a polygon. After digitizing, the data was topologically structured and polygons were formed.

All data derived from separate and adjacent map sheets were edge-matched to ensure topological and attribute continuity. After edge-matching, the maps were merged to produce one seamless map of the study area.

After merging, all data were rubber-sheeted to known SPC's. State plane coordinates for section corners on the RMA were obtained from a surveying firm and then compared to the section coordinates determined through digitizing. The database was then rubber-sheeted according to a least squares transformation to yield a more accurate coordinate database across the entire study area.

Database Analysis

Once the database was complete, several data themes were derived using cartographic analysis techniques; reclassification, overlay, proximity, and compositing.

The map database was reclassified according to geographic viewing windows and primary attributes. For example, hydrographic and man-made features data were combined to create an exclusion area theme.

Overlay analysis was performed between topographic contours and ground water contours to produce a depth-to-ground water map and also between topographic contours

and bedrock contours to produce a depth-to-bedrock map. These products were then overlaid to determine difference between groundwater and bedrock depth and produce a saturated alluvium theme (i.e., ground water elevation higher than bedrock elevation).

Proximity analysis was performed to determine 1000', 0.5 mile, and 1 mile buffers inward from the RMA boundary.

Compositing (map merging) was used to produce final map displays of more than one data theme.

For editing and site selection previewing, hard-copy plots and interactive graphics monitor displays were developed. For each map product, a legend identifying attributes and codes; a title identifying participating agencies and companies, map source, map number, and approvals; and a map title, bar scale, and north arrow were added. Then colors were assigned to lines and text, and solid fill or shading assigned to polygons. Pen plots in black and white, and color, were produced on paper and mylar at different scales. Graphic monitor displays were produced interactively and also saved as "slide" files for faster display at a later time.

The Cartographic Model

A GIS lends itself to the solution of cartographically related problems through construction and solution of a cartographic model. Typically, the desired end product is identified first. Next, the steps and data required to reach this end product are identified. Thus, a tree-diagram or flow chart is constructed from final product (left) to beginning products (right) and then a GIS executes this model from right-to-left.

The end product was to identify a potential landfill site for hazardous waste disposal on the RMA. Siting criteria included:

- a maximum depth-to-ground water
- a minimum depth-to-bedrock
- at least 1/2 mile inwards from the RMA boundary
- location outside of the 100-year floodplain
- a minimum size of 1,000 acres
- away from existing infrastructure (roads, pipelines, buildings, etc...)

Several intermediate map products were necessary to reach a decision on a proposed site and were derived through analysis, as described previously.

RESULTS

As the cartographic model was executed,

interactions among the siting variables necessitated a trade-off among criteria to define multiple potential sites. Interactive computer sessions were held with technical siting experts and executive decision makers and alternatives and trade-offs in the criteria were tested.

Six potential sites were defined during these interactive sessions. Criteria which were varied included distance from the RMA boundary, depth-to-groundwater, and depth-to-bedrock. Criteria which were deemed to be inviolate included floodplain, exclusion area, and site acreage. Users then ranked each site based upon their assessment of the priority of each criteria.

Once potential sites had been identified the GIS "slide" function was used to save the screen displays for future high-speed viewing by other decision makers. Thus, during interactive computer sessions, executive decision makers were able to identify and choose between site alternatives. Using traditional engineering drafting methods, this process might have taken months. Use of the interactive GIS slide function allowed rapid viewing of alternatives and recommendation of the preferred potential hazardous waste landfill site.

When this project began in May 1986, STRINGS was one of the only PC-based vector GIS' with an arc/node data structure facilitating complex polygon overlay analysis. Other PC-based GIS' are now available. While this paper is not an endorsement of, or a comparison between, systems, several lessons were learned on this project. The STRINGS package proved to have an efficient and user-friendly data entry, analysis, and display system. The database for this project had to be compiled, edited, and analyzed, and display products created within 2 months. Use of STRINGS facilitated this task. All project deadlines were met and executive decision makers were generally pleased with output products.

The slide function of the STRINGS GIS proved particularly useful in interactive sessions. This function allows screen graphics displays to be built and then stored as an image file for rapid viewing at a later time. Thus, complex displays could be created by an operator and then viewed by decision makers without having to wait for graphics processing to be accomplished by the GIS. These images were repeatedly displayed to compare pros and cons of alternate sites. If new slides were requested during an interactive session, executive decision makers would take a break while analysts prepared a new slide file for review. This new slide would then be compared with previous slides until decision makers were satisfied that all site alternatives had been evaluated and that trade-offs between sites were fully understood. In this

way, a preferred site was determined in 3 1-hour interactive computer sessions versus months of manpower effort which would be required using traditional engineering drafting methods.

CONCLUSION

This paper has attempted to demonstrate how a GIS can be used to identify potential landfill sites for hazardous waste disposal. By utilizing STRINGS, geographic data was effectively captured, stored, edited, analyzed, and displayed using an IBM AT-based GIS. Here, base map data on the RMA was captured digitally, analyzed through a cartographic model, and finally the map products and cartographic model solutions were displayed in plots and screen images. Use of interactive computer sessions allowed executive decision makers to rapidly identify alternatives and trade-offs and further refine the cartographic model.

TESTING LARGE-SCALE DIGITAL LINE GRAPHS AND DIGITAL ELEVATION MODELS IN A GEOGRAPHIC INFORMATION SYSTEM

by

David R. Wolf
U.S. Geological Survey
521 National Center
Reston, VA 22092

E. Terrence Slonecker
The Bionetics Corporation
P.O. Box 1575 V.H.F.S.
Warrenton, VA 22186

ABSTRACT

In February 1988, the U.S. Geological Survey and the U.S. Environmental Protection Agency (EPA) entered into a cooperative investigation of the use of geographic information system (GIS) technology in the CERCLA (Superfund) Remedial Investigation process. EPA has over 28,000 sites in various stages of this process and is investigating mechanisms that can efficiently analyze the large amounts of spatial data that are associated with Superfund site investigations. The Old Southington Landfill in Southington, Connecticut, was chosen as a pilot site. This site is currently in the Remedial Investigation/Feasibility Study stage of the Superfund remedial process.

To evaluate the landfill's potential for contaminating the surrounding environment, a large-scale GIS data base was created. The data base included custom Digital Line Graphs (DLG's) generated from a digital analytical stereoplottor and coded in standard DLG format. Also under evaluation in this project were custom Digital Elevation Models (DEM's) and a unique site-feature data set compiled from historical aerial photographs.

Several application scenarios were tested and the results presented at EPA's Remote Sensing/Technical Support Symposium in May 1988 to demonstrate the advantages of incorporating remote sensing and GIS technology into the Superfund remedial process.

INTRODUCTION

Geographic information is of vital importance to the role of government in general and specifically to the mission of the Environmental Protection Agency (EPA). Most scientific disciplines are in some way concerned with spatially distributed phenomena and EPA studies nearly always involve multidisciplinary approaches. In The Comprehensive Environmental Response, Compensation, and Liability Act (CERCLA) and The Resource Conservation and Recovery Act (RCRA) site investigations, the majority of the data generated in the Remedial Investigation

Publication authorized by the Director, U.S. Geological Survey.

Any use of trade names and trademarks in this publication is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

process is spatial in nature and is derived from such diverse sources as monitoring wells, utilities maps, political boundaries, ecological data, census tracts, and airborne remote sensors. The ability to process and analyze spatial information is central to the mission of the agency.

A geographic information system, or GIS, is a system of computer hardware, software, and procedures designed to store, analyze, and display spatial information. Spatial information is any information that can be mapped, or referenced "geographically." GIS technology has given us the capability to integrate and analyze large amounts of spatial data that would not have been possible with analog techniques. GIS technology has emerged in recent years from the realm of research and development to one of application and is now rapidly becoming a new and powerful tool for integrating and analyzing spatial data.

The EPA has been investigating the use of GIS in various mission-related applications for several years. One such mission, Superfund site analysis, has been selected by the EPA's Advanced Monitoring Division (AMD) for evaluation by a GIS. This report will not deal with the complex modeling and site analysis that a potential hazardous waste site might be subject to during the CERCLA process, although GIS could assist those operations as well, but will concentrate on the design, production, and application of large-scale, site-specific digital cartographic structures.

EPA has long endorsed the use of historical imagery to fully examine the chronology of hazardous waste sites. Such a study has been done for the Old Southington, Connecticut, landfill. In order to more efficiently utilize this investigation, a pilot GIS demonstration was developed at the U.S. Geological Survey (USGS) incorporating spatial data from EPA, USGS, Soil Conservation Service (SCS), the State of Connecticut, and other sources.

The demonstration employed large-scale custom Digital Line Graphs (DLG's) in a GIS. The DLG is a standard digital mapping product of USGS and portrays spatial themes such as transportation, hydrography, and cultural features. However, at a scale of 1:24,000, the largest production DLG scale available, the standard product lacks sufficient detail and spatial resolution to address the thematic elements of information that are essential to large-scale, site-specific analysis. Additional spatial features, identifying site characteristics not found in standard DLG coding, needed to be implemented to assess the potential for contamination from the landfill. Also, large-scale Digital Elevation Models (DEM's) were compiled to better view the physical relief of the landfill during its operation.

THE REMEDIAL PROCESS

The CERCLA not only established funding mechanism for the cleanup of hazardous waste sites but also defined procedures for the study and evaluation of remedial options. To effectively study the complex issues surrounding most hazardous waste sites, a comprehensive strategy for data collection, processing, testing, sampling, and evaluation is required. This strategy is known as the Remedial Investigation/Feasibility Study. The final decisions must weigh the need to safeguard public health and environmental quality at a specific site against the ability to fund the process there and at other sites across the country.

The instrument of data collection in the remedial process is the Remedial Investigation and the instrument of analysis is the Feasibility Study. The Remedial Investigation emphasizes site characterization and investigation, while the Feasibility Study is directed at weighing options and providing the necessary analytical tools for making decisions.

SITE HISTORY

The Old Southington Landfill is a former municipal landfill located along Old Turnpike Road in Southington, Connecticut. Between 1920 and 1967, the landfill was utilized for the disposal of residential, commercial, and industrial wastes in both liquid and solid form. In 1971, the Town of Southington installed a municipal water well (No. #5) approximately 700 feet northeast of the landfill. In 1971, the well was closed due to elevated and unacceptable levels of trichlorethene, a common industrial solvent, which exceeded the Connecticut Department of Health water quality standards. Collateral data from the EPA and the State of Connecticut indicate that several types of industrial wastes, including those in drums, were accepted into the landfill during its period of operation. The site was predominately a wetland area prior to its becoming a landfill.

The closure of municipal well No. 5, in addition to two other municipal wells within the area, prompted EPA to request a historical aerial photographic analysis of the area by its AMD. To inventory past and present potential contamination sources within an approximately 3-kilometer radius of the closed wells, AMD asked its field station, the Environmental Photographic Interpretation Center (EPIC), to research, acquire and analyze all relevant historical aerial photography. Following the completion of this inventory study in January 1984, EPIC was requested to conduct a more intensive, site-specific aerial photographic analysis of the Old Southington Landfill. This report was completed in February 1988. The landfill is an EPA Superfund site and a potential responsible party in the contamination of well No. 5.

In March 1988, the landfill was selected for a pilot study to demonstrate the applicability of integrating remote sensing and GIS technology to support the acquisition, generation, and processing of site information essential to the Superfund remedial process. Information from the various remote sensing studies along with other relevant data generated by EPA, the State of Connecticut, USGS, and the Southington Chamber of Commerce were used to develop various data sets, models, and scenarios that would relate directly to the needs and requirements of the Remedial Investigation process.

LARGE-SCALE DLG/DEM PRODUCTION

One of the major drawbacks encountered in creating a large-scale GIS data base is the lack of large-scale digital map data. The largest scale data that is commonly available throughout the country is the 1:24,000-scale quadrangles of the USGS. However, for site-specific work, this scale is inadequate for the type of detail and accuracy that is necessary in Remedial Investigation activities. Often the only alternatives are to digitize an existing map of sufficient scale and detail or create a new one. Because there was no existing map deemed suitable by the GIS team, one was photogrammetrically created using a digital analytical stereoplotter and a quad-based control network.

The instrument used to produce the map was an Intergraph Corporation InterMap Analytic (IMA) photogrammetric workstation. The IMA is a first-order instrument that utilizes advanced analytical stereoplotter technology with interactive graphic capabilities that allow the operator to digitize, code, and create digital data structures in an interactive, three-dimensional environment.

Aerial photography acquired in 1986, as part of the standard historical site investigation was utilized for map compilation. The black-and-white photographs were standard 9 by 9 inch format and were acquired according to mapping cartographic specifications. The scale was approximately 1:12,000. Because time constraints did not allow for the establishment of surveyed control data, control was generated by digitizing selected photoidentifiable coordinates from a stable-base, 1:24,000-scale USGS color separate. The photogrammetric model was set using this control and followed standard model setup procedures (interior, relative, and absolute orientations). The final root mean square error for the production model was approximately 5 meters horizontally and 0.3 meters vertical.

Because the temporal aspects of this site spanned a considerable amount of time, the surrounding land use depicted on a current map base would not be sufficient for portraying historical development. To solve this problem, a second DLG was created from 1951 imagery to show the differences in the surrounding land use and to provide a more realistic base for the historical thematic overlays.

The DEM's were produced by using an OMI AS11A1GS analytical stereoplotter and standard profiling software developed by USGS. To capture the subtle changes in landscape and terrain that would be essential to large-scale applications, the sampling interval was reduced to approximately 5 meters instead of the usual 30 meters that is standard in DEM production. As with the DLG's, both 1986 and 1951 aerial photographs were used to create the data that showed the terrain profile differences in a historical setting.

CREATION OF A LARGE-SCALE CUSTOM DLG AND DEM DATA BASE

After processing the newly compiled large-scale topographic data set, the data were converted for entry into the GIS data base. Attribute information corresponding to the standard USGS DLG product was assigned to the topology and built into the relational data base management system. These attributes include topographic features such as roads, trails, streams, and wetlands. In addition, buildings were delineated and attributed as to their use, such as residential, commercial, or light industrial.

Because of the historical nature of the investigation, 1951 photographs were used to create an additional DLG and DEM to help analyze activities at the site during the course of its operation. The attribute coding for the historical DLG was identical to the present-day DLG.

This data set helped to reflect changes in the topographic, land use, and wetland characteristics found around the site while the landfill was active.

Once entered into the GIS, basic area and linear measurements were automatically calculated from the topology and became a part of the data base management system. The large-scale study area containing the custom DLG is approximately one square mile, while the full extent of the landfill is 15.75 acres. A quick GIS

analysis of the two custom DLG's reveals that the closed landfill now contains four residential buildings and six commercial/industrial buildings and contains approximately 10 acres of wetland.

CREATION AND ANALYSES OF A SITE-FEATURE DATA BASE

Site features, compiled from historical aerial photographs, have been digitized and coded in the same manner as the topographic DLG's. Using the same instrument to produce the site-feature DLG, a data base of specific features associated with potential contamination sources was created to help document past activities attributed to the landfill. These potential sources of contamination were recorded from photography flown in 1941, 1951, 1957, 1965, 1967, and 1970, covering the entire existence of the landfill.

These features are especially important in developing data for the site history and characterization needs of the Remedial Investigation.

Because the standard DLG coding scheme relies on a unique set of feature attributes, a cross-over coding classification scheme (table 1) was implemented to take advantage of the standard DLG encoding system during digital compilation of the site-feature DLG.

A site-feature DLG was created for each year recorded during the photointerpretation phase of the EPIC investigation. A test was devised that would help identify the positions of these historical site features in relation to present-day topographic and cultural features. By selecting significant features or features that would automatically raise questions as to their potential threat to the surrounding environment and overlaying them with present-day topographic and cultural DLG's a graphic was produced that might produce questions and answers regarding the Remedial Investigation.

For example, the four homes in the extreme north of the landfill are built near a 5-acre debris field. By examining the historical site-feature DLG, it is noted that the debris field was active during the mid to late 1960's. Does the old debris field pose a health threat to the occupants of these homes? Another related issue might be, did the home owners know of the proximity of the debris field before their purchase? Farther south, within the landfill, a trench containing standing liquid appears on the 1951 site-feature DLG. This signature often indicates improper disposal methods and is usually investigated for a possible threat to the ground water. By observing the position of an existing building, the placement of a monitoring well or core sample site, which could determine the identity of the liquid, might be better located. Off site and northwest of the landfill, the 1970 site-feature DLG reveals a small area of less than a half acre containing drums, debris, and mounded material. This observation by itself might not be cause for great alarm, however, given the area's proximity to a known contaminated well (less than 100 meters) and the fact that an industrial building sits on top of this potential hazardous waste site, questions arise. Were these materials properly disposed of or are they now part of the present building's foundation?

Table 1.--Digital Line Graph cross-over coding classification scheme epic legend/DLG crossover coding - project pic 88084

[Environmental Monitoring Systems Laboratory Report TS-PIC-88031, site analysis, Old Southington Landfill, Southington, Connecticut]

EPIC ¹ LEGEND (MEANING)	DLG CODE	(MEANING)
Auto junkyard	200.0423	(oil reservoir)
B (building)	200.0402	(church)
Berm	200.0211	(coke ovens)
C (containers)	200.0403	(school)
Cleared area	200.0406	(Post Office)
DB (debris)	200.0404	(municipal Building)
DK (dark-toned)	200.0601	(underground)
Discolored	200.0607	(chemical)
Drums	200.0451	(swimming pool)
E (equipment)	200.0405	(courthouse)
EX (excavation/extraction)	200.0430	(strip mine)
Fill area	200.0427	(mine dump)
GS (ground scar)	200.0445	(fairgrounds)
Graded area	200.0452	(ruins)
IM (impoundment)	200.0453	(recreation area)
LT (light-toned)	200.0603	(abandoned)
M (material)	200.0434	(storage bin)
MM (mounded material)	200.0436	(spoil bank)
Open trailer	200.0410	(town)
OS (open storage)	200.0450	(fort)
Pit	200.0432	(pit)
Pond	200.0421	(STP)
REV (revegetated)	200.0608	(covered)
Revegetated area	200.0447	(corral)
SL (standing liquid)	200.0433	(radio/TV facility)
ST (stain)	200.0454	(picnic area)
TR (trench)	200.0465	(pile, dolphin)
UC (under construction)	200.0602	(under construction)
V (vehicles)	200.0468	(sunken wreck)
Well	200.0307	(drilled well)
SYMBOLGY ONLY		
----- Access road	200.0200	(conveyor)
- Drainage	200.0201	(broadwalk)
_____ Site boundary (solid line)	200.0206	(fence)
- - - - - Historical boundary (long-short/long-short)	200.0422	(waterworks)
_____ Sloped edge : : : : : :	200.0435	(levee)

¹ Environmental Photographic Interpretation Center, Environmental Protection Agency

PROPERTY OWNERSHIP

An important part of the Remedial Process is the determination of both the responsible parties and those potentially affected. As in many multipurpose cadastre issues, the importance of exact locations of property ownership boundaries is of paramount concern. For a quick assessment of property ownership, however, a property data base was extracted from county tax records and digitized and transformed to the baseline coordinates of the custom DLG.

This process allows an investigator to look at the chronology of ownership during the life of the landfill and to relate specific site characteristics to specific years of ownership. Because only those tax records that related to the years of photointerpreted site characteristics were used in the analysis, the results were not always be conclusive, but they helped to identify areas that required more information.

The property ownership investigation focused only on the landfill as it was identified in the photointerpretation phase. This meant that ownership boundaries were compiled for the landfill but not for surrounding areas. Even though early analysis of the actual landfill indicates less than full use of the area delineated as the landfill extent, all property boundaries within the site are defined. Also, to simplify site characterization within the property lines, a significant feature data set was extracted from the overall site-feature data base.

The initial photo analysis of the site utilized 1941 imagery that coincided with the beginning of activity in the landfill. Although only an isolated debris field is revealed, it is well within the property lines of the Town of Southington, operators of the landfill. Ten years later, 1951, a slight movement south of the initial debris field is evident; however, all landfill activities remain within the town's property lines. Analysis of the 1957 data reveals the first significant features, mounded material/stains, to be found off the town's property. Also, the development of an elongated debris field with a north/south orientation and curving west to stay within the town's property boundary is evident. The 1965 ownership/site-feature analysis produces some new ownership and possibly some new strategy in landfill debris collection. All major debris accumulation now occurring off the town's property, and the largest debris field, 1 acre, is located within the adjacent property to the north.

The change in location of landfill activity also coincides with a change in ownership of the northern parcel. Expansion of the debris field to 1.5 acres is detected within the northern parcel 2-years later, 1967, while landfill activities show signs of decreasing within the town's property lines and elsewhere. Termination of landfill activities is complete in 1970, although associated activities offsite have picked up, possibly because of the closing of the landfill.

DIGITAL ELEVATION MODELS APPLICATIONS

Large-scale DEM's were created from 1951 and 1986 aerial photographs to address several issues that typically arise in the Remedial Investigation. DEM's were created for the same square mile area that was mapped during the large-scale DLG production. After transforming the DEM data into ARC/INFO Lattice and triangulated irregular network formats, several application scenarios were developed.

First, the overall area was transformed into three-dimensional perspective views showing the topographical relationship between the landfill and the surrounding area. These perspectives not only give the viewer a better understanding of the topography of the overall

area but are also detailed enough to show subtle drainage and runoff characteristics. When the DLG and other thematic overlays are draped over the three-dimensional image, the interrelationships among the various information elements can be better understood. This type of interrelated information is critical to network modeling, contaminant migration studies, and risk assessment, which are key issues in the Remedial Investigation process.

Second, the landfill boundary is extracted from the large-scale DEM for both 1951 and 1986, and volume calculations were performed for each year. The difference between the 1951 and 1986 volume theoretically shows the total amount of fill material that has been placed in the landfill during that period, which is roughly before and after landfill operations. This information can be used in a variety of Remedial Investigation scenarios, including cut-and-fill and other engineering applications.

Finally, an individual property was extracted from the DEM and the same volume calculations performed for that property. This information can be utilized as a quantifying factor for damage calculations and risk assessment.

SUMMARY

This pilot project demonstrated that custom large-scale DLG's and DEM's can be generated for use in a GIS. For those GIS applications that do not need custom baseline data sets existing DLG's, might be appropriate. However, if the requirement exists, such as in this Superfund site investigation, the capability to create the required unique data base is available.

In this case the need for custom large-scale data sets was directly related to the application. When areas to be examined are less than a few square miles and precise measurements are required, topographic and other features must be precisely positioned and existing data sets might not always be suitable for this purpose.

Concerning the issue of spatial accuracy USGS complies with National Map Accuracy Standards for its graphic maps and sets its digital standards from these stable-base cartographic products. If larger scale DLG's were systematically produced for use in analytical investigations, such as the one described here, then accuracy and coding standards must first be addressed. Questions regarding cartographic data bases, such as the USGS DLG, versus geographic data bases compiled from original source material, such as the custom DLG used in this project, are being addressed by USGS and other organizations.

As a result of the Old Southington GIS project, EPA and USGS are investigating accuracies and standards that will ensure the integrity of spatial data and multipurpose cadastre analysis of the remedial process. In December 1988, a Global Positioning System (GPS) survey was performed at the landfill site to determine the feasibility of using this technology for remedial investigations. The GPS survey will provide precise coordinates for stereocompilation of the large-scale DLG. A statistical spatial accuracy test is being performed as part of ongoing interagency research and will be reported on at a later date. The EPA has also initiated a study to standardize a coding scheme, such as the cross-over classification system used in this project.

REFERENCES

Goldberg-Zoino & Associates Inc., 1987, Work Plan for Remedial Investigation/Feasibility Study at the Old Southington Landfill Study Area, Old Turnpike Road, Southington, Connecticut: File Number H-50124.01, Bridgeport, Connecticut.

Sitton, M.D., 1988, Site Analysis, Old Southington Landfill, Southington, Connecticut: USEPA Report TS-PIC-88084, Warrenton, Virginia.

Techlaw, Inc., 1987, Draft Property Report, Old Turnpike Road Landfill, Southington, Connecticut, Responsible Party Search; Contract No. 68-01-7331, Boston, Massachusetts: USEPA Report EPA/540/G-85/002, Washington, D.C.

U.S. Geological Survey, 1986, Large-Scale Mapping Guidelines: U.S. Geological Survey Open File Report 86-005, 47 p.

Walsh, S.J., D.R. Lightfoot, and D.R. Butler, 1987, Recognition and Assessment of Error in Geographic Information Systems: Photogrammetric Engineering and Remote Sensing, Vol. 53, No. 10, pp. 1423-1430.

QUADTREE MESHES

William T. Verts
COINS Department
University of Massachusetts
Amherst, MA 01003

Professor Francis S. Hill, Jr.
ECE Department
University of Massachusetts
Amherst, MA 01003

ABSTRACT

Quadrees have long been a favorite data structure for reducing the memory storage requirements of bilevel images and for representing those images hierarchically. In general, a quadtree requires far less storage than the corresponding unencoded image. Unfortunately, storage requirements depend critically on the offset of the image within its sampling grid; quadtrees are variant with respect to translation. Reducing the amount of storage required by a quadtree implementation is strongly related to reducing its sensitivity to translation. Techniques that address these issues include Linear Quadrees, Quadtree Normalization, the Quadtree Medial Axis Transform, and Quadtree Forests. The Translation Invariant Data structure is a related non-Quadtree technique based on medial axis transforms. This paper presents a translation invariant representation that maintains both the hierarchical properties and spatial coherence of each object in an image. Each image object is allocated its own quadtree, then those objects are interconnected with a meshwork (such as a Delaunay Triangulation) based on object centers. The geometry of the meshwork allows each object to be translation independent of all others (they may overlap), and allows the composite image to be of arbitrary size. The properties of this technique are explored and applications to cartography and extensions are discussed.

INTRODUCTION

As reviewed by Samet (Samet; 1984), there are several types of quadtrees with different properties and applications. For this paper we focus on the most widely known type, the *region quadtree* (hereafter referred to simply as a "quadtree").

What is a Quadtree?

Quadrees are tree data-structures useful for storing bilevel images. Extensions that allow quadtrees to capture gray-scale images will be considered later. An image to be represented by a quadtree is an N by N square array of pixels, where N is a power of two ($N = 2^L$, for some positive L). Each pixel may be either *black* or *white*; black pixels represent part of an image object, white pixels are background. Following convention, we call the two levels black and white, but in general the image element may have any two discrete values.

Each node in a quadtree is either a leaf node or an internal node. Leaf nodes of the quadtree correspond to a region of the image that contains a single color: all pixels are black or all pixels are white. Internal nodes, with pointers to four descendants, represent regions that are a mix of black and white pixels. The region covered by any internal node is the union of the four regions covered by its descendants (a perfect tiling; there are no gaps and no overlap). Other auxiliary information may also be stored in each node, such as a pointer to its ancestor, or some statistical information about the image in its subtree (a *gray* level representing the average color of the region covered by the node, for example).

How is a Quadtree Formed?

Most images contain some mixture of black and white areas. If the image is entirely black or entirely white, the quadtree captures the entire image with a single leaf node. Otherwise, it is divided into quarters, where each quarter corresponds to one subtree of the quadtree root. Each of the four descendants of the root corresponds to a quadrant of the image, called the NorthWest, NorthEast, SouthWest and SouthEast quadrants, respectively. The process is recursively repeated on each quadrant until the subimage contains a single color. See Figure 1.

Attractive Properties of Quadtrees

If the recursive decomposition of the image stops before the pixel level is reached, the corresponding quadtree will typically require less storage than that required by the original (uncompressed) image. Local coherence in the image tends to reduce the number of unique points that are stored.

Quadtrees also encode images hierarchically; if gray information is stored at internal nodes, coarse approximations of an image can be constructed by examining all nodes at a single level in a quadtree; every level covers (tiles) the entire image area. More detail appears as deeper levels of the quadtree are examined. When a branch of the quadtree ends in a leaf the size and color of that leaf are used in all lower levels.

Problems With Quadtrees

Normally, there are several *objects* in the scene captured by the image. The number of nodes required by a quadtree can change drastically by shifting those objects within the image (translation variance). Adding points to and deleting points from an image can also drastically change the number of nodes in the corresponding quadtree.

Image objects have no internal *coherence*, and may be split among several quadrants during the formation of the quadtree.

White areas are explicitly stored in the quadtree. Image objects are fully described by the black nodes of the quadtree; the white nodes are redundant. Techniques to eliminate or reduce the number of white nodes are outlined in the next section, but in general some must be stored.

It is extremely difficult to add points to an image outside of the (2^N by 2^N) sampling grid. If there is space available the image can be shifted over to allow the new points to be added. Shifting will not be possible in all cases; when it is possible the quadtree will be affected by translation variance.

Some images can not be easily compressed. In cases where every pixel must be present in the quadtree (as in a checkerboard) more storage space is required than that by the uncompressed image, due to the overhead of the quadtree structure. In a checkerboard every pixel differs from its neighbors; no spatial optimization can take place.

ALTERNATIVE METHODS

Several methods have been proposed to reduce both storage and translation variance in representing images.

LINEAR QUADTREES

One method used to reduce the overhead associated with quadtrees is to store them as Linear Quadtrees (Gargantini; 1982). A linear quadtree is a pointer-less representation that stores only the black nodes of a quadtree. There is one entry in the linear quadtree for each black leaf in the full quadtree; each entry is a coded path from the root to the leaf. The symbols in the coded path are in a "quaternary" (base 4) code, where symbols

0, 1, 2, and 3 stand for the NW, NE, SW, and SE descendants of a node, respectively. The number of symbols along the coded path corresponds to the depth of the leaf in the quadtree. The linear quadtree for the image in figure 1 is: 00, 010, 012, 020, 021, 03, 10, 12, 203, 21, 30.

Linear quadtrees have some nice properties. They can be transmitted via a text-only link because of the ease in which the quaternary codes can be converted into ASCII characters. They can readily be expanded back into their original images, and it is easy to perform Boolean compositing operations (union, intersection, etc.) between two linear quadtrees.

QUADTREE NORMALIZATION

Quadtree normalization is a technique used to reduce the number of nodes in a quadtree by shifting the image around in the image grid. It has been shown (Li, Grosky, Jain; 1982) that the optimum placement of an N by N pixel image can be computed in time $O(N^2 \log_2 N)$. Their algorithm takes an image and returns X and Y shift factors to obtain the optimum (minimum) number of quadtree nodes. The optimum placement of the image within the grid may require that the grid size be doubled (quadrupling the grid area).

Normalization is also used in (Chien, Agarwal; 1983) to simplify the recognition of a class of objects (jet planes in their paper) that may be in any size, placement, or orientation on an image grid. By normalizing their list of expected objects with respect to size, principle axes and centroids, they use the quadtree representations as "shape descriptors" to perform pattern matching. Note that the entire image is not converted to a quadtree, only objects within the image. This "decoupling" of objects from their background is an important tool used in the Quadtree Mesh technique described below.

QUADTREE MEDIAL AXIS TRANSFORM

Samet (Samet; 1983 & 1985) describes a technique for transforming one quadtree into another quadtree with fewer nodes (additional information must be present in each node, however). The new quadtree contains both black and white nodes as before, except that associated with each black node is an integer *radius* which indicates the size of the black square. The square defined by the radius may or may not cover a larger area than the subtree node would otherwise cover. The radius will never be smaller than the "natural" size of the black square. As the radius grows larger and larger, fewer and fewer nodes must be kept in the tree structure.

The radius may indicate that the corresponding square extends beyond the borders of the image, and for this reason it is assumed that all space outside of the image boundary is black.

Asking if a particular pixel is white or black involves more than simply traversing the tree; a white node in the quadtree may be white, but it may also be covered (partially or totally) by a nearby large-radius black node. Neighboring cells in the tree must be examined to see if there is any overlap.

The worst possible outcome for the Quadtree Medial Axis Transform (QMAT) is that the resulting quadtree will be identical to the original quadtree. The result will never contain more nodes than the original, but it often contains significantly fewer nodes. This also means that a QMAT tends to be less sensitive to shift than its quadtree counterpart.

QUADTREE FORESTS

In (Jones, Iyengar; 1981) an idea is presented that offers space savings by breaking a quadtree into a list (or forest) of trees where each tree is a small section of the original structure.

A normal quadtree is labeled according to a simple recursive algorithm as follows: any black leaf is considered to be *good*; any internal (gray) node with two or more good descendants is also considered to be good; all other nodes are *bad*.

Once the quadtree has been labeled, the forest is formed by detaching subtrees at their highest good point from the labeled structure. If the root node has been labeled as good, the result is a forest of a single tree (the original structure).

Each entry in the forest contains a pointer to the section of the tree that has been labeled as good, and also the level and path information that serves to position the tree section within the original image. The path key is very similar to the quaternary codes of the linear quadtree.

This technique does not eliminate white nodes from the resulting tree structures, but it does tend to reduce the amount of white space that is stored.

The worst possible outcome for a quadtree forest occurs when the image is a kind of checkerboard where each and every 2×2 pixel area contains at most one black pixel. The quadtree is fully developed to the pixel level, and the corresponding forest consists of one entry for each (1×1) black pixel (Gautier, Iyengar, Lakhani, Manohar; 1985).

This technique suffers from many of the same problems that normal quadtrees suffer from; they are shift variant and cannot take advantage of object coherence within an image. Shifting an image within an image grid will require a new forest to be developed. The list of subtrees is also presented as a linear list, with no topological relationships between the locations of trees in the forest.

COMPACT QUADTREES

In (Jones, Iyengar; 1983) one more technique is developed to reduce the amount of storage with a standard pointer-based quadtree. Instead of containing in each node one color field, one pointer for each of the four descendants (all Nil in case of a leaf node) and one pointer for the ancestor, each *metanode* contains one ancestor pointer (MFATHER), one descendant pointer (MSONS), one brother pointer (MCHAIN), and the colors of all four quadrants; black, white, or gray. A gray value in the color list implies that there is a metanode attached to the descendant pointer. If more than one gray appears in the color list, the additional metanodes will be chained via the brother link off of the descendant node.

The combination of keeping all four colors and fewer pointers in each node reduces both the number of nodes required and the overall number of pointers.

TRANSLATION INVARIANT DATA STRUCTURE

The final technique for reducing the size of an image representation is not a quadtree technique at all, but it has some attributes that make it worthy of discussion. The Translation Invariant Data Structure (TID) (Scott, Iyengar; 1986) is based on the medial axis transform of an image, and operates by breaking the image down into a list of black maximal squares along with their location and radius. The squares may or may not overlap, and the union of all squares is the original image.

A TID is translation invariant by virtue of the fact that the locations in the list can be considered to be relative to some origin; change the origin and the objects in the image have moved, but without disturbing the relative positions of the objects. The locations in the list can also be modified according to which of several objects is moving relative to the others.

The storage requirements for a TID are no worse than for any of the quadtree techniques outlined above, and may be considerably better. Forming a TID from a

region that is R rows by C columns is of the order $O(RC \log(\min(R,C)))$ (Scott, Iyengar; 1986). See also (Gautier, Iyengar, Scott; 1985).

One drawback of using TID is that any hierarchy of subimages or of objects is lost. No square has any priority over any other, and no coarse resolution images can be quickly extracted as in the case of the quadtree (or any of its variants). This technique might also be extended to use rectangles instead of squares for obtaining better matches of the objects being captured.

QUADTREE MESHES

The previous techniques all address one or more of the problems associated with quadtrees. What we present here is a synthesis of several of the aforementioned techniques, with some new considerations thrown in.

What is a Quadtree Mesh?

A Quadtree Mesh is a collection of quadtrees that may be placed anywhere in the plane, with a geometrical meshwork (graph) applied to the origins of the quadtrees. The quadtrees may overlap, and may contain conflicting information about a particular subregion.

In general, each quadtree contains the image of a single object or of a group of objects that belong together (figure 3). These *object quadtrees* are connected together with a meshwork. The meshwork may be as simple as a linear list (with all its inherent disadvantages) or it may be some form of optimal mesh such as a Delaunay Triangulation (figure 2).

Each entry in the mesh contains the following information: a pointer to the quadtree itself, the power of two that describes the side length of the square area covered by the quadtree, and the *image origin* (coordinates of the upper left corner of the square, for example). Additionally, it is very useful in geographic applications to keep the coordinates of the primary *point of interest* of the area, relative to the origin point. This reference point can be the origin point itself, the image center, the center of mass of the image object, or even some point completely outside of the image. In figure 2, the reference point in each object is the NorthWest pixel of the four pixels that surround the object center. The positions of the reference points are used when deciding how the quadtrees are to be connected together in the mesh.

Quadtree mesh images may be as large as necessary: because the origin and reference points may be anywhere in the plane, overall image size is bounded only by the integer precision and memory capacity of the computer system being used. The size of an image area is stored with each object quadtree, and that area can be as small as a single pixel or as large as the entire integer plane. The number of levels in each quadtree (its depth) can be easily determined from its size.

A quadtree mesh applied to a bilevel image effectively partitions that image into three values; black and white (inside at least one object quadtree), and *unknown* (outside all object quadtrees). For most purposes, unknown can be considered equivalent to white.

Advantages of a Quadtree Mesh

Quadtree meshes eliminate the problem of variance with respect to translation. Moving an object from one place to another entails changing only its origin point and does not affect the structure or contents of the corresponding object quadtree.

Each object quadtree need be only as large as necessary to capture its image object; although there is some white space stored, that amount of white space is relatively small. The implementation of the object quadtrees could be as Quadtree Forests, Normalized Quadtrees, QMATs, Compact Quadtrees, or any other method that reduces

the overhead of storing a single quadtree. Each object quadtree could use a different method of storing the image depending on which method optimizes the image object best. The entries in the quadtree mesh would then need a field describing which of the methods is used in the corresponding object quadtree.

Four objects are shown in figure 3, partitioned according to how they would be described with quadtrees. Objects 1 and 2 are 4x4 regions that require object quadtrees of one leaf each (because object 1 is identical to object 2, only one copy needs to be constructed; the corresponding mesh nodes can share a single instantiation of the quadtree). Object 3 is a 4x4 region that requires an object quadtree with 10 leaves, and object 4 is a 7x7 region (normalized to the lower right corner of its 8x8 square) that requires an object quadtree with 40 leaves. Together, regardless of their placement within an image, the four object quadtrees require 52 leaves (51 if the quadtrees for objects 1 and 2 are shared). The two 16x16 images in figure 2 contain the four objects in different places. A quadtree describing the left image requires 142 leaves, and a quadtree describing the right image requires 103 leaves.

Searching

When looking for the color of a particular pixel, a search uses the meshwork to find the cluster of interest. The configuration of the meshwork can simplify many problems in geometry.

If object quadtrees describe areas of similar sizes, then finding the “correct” quadtree entails traversing the mesh to find the closest point of interest to the search pixel. If a Delaunay Triangulation is used as the mesh geometry, then finding the closest point is trivial: choose any point in the mesh as the current point, examine all points that neighbor (are connected to) the current point and set the current point to the one closest to the search pixel, and repeat until there are no closer points. The quadtree associated with the resulting mesh point is searched for the pixel value. By the properties of Delaunay Triangulations, the resulting mesh point will be one of the three points on the perimeter of the triangle enclosing the search pixel if the pixel is inside the convex hull of the mesh space, and it will be the closest hull point if the search pixel is outside. More than one object quadtree may contain the pixel, and a conflict between quadtrees may have to be resolved. Conflict resolution is addressed in the next section.

Once a pixel has been found, searching for adjacent pixels is fairly straightforward; if the new pixel is inside the current (just searched) quadtree, the same quadtree can be used. If it is not in the current quadtree, it may be in a “nearby” quadtree. The mesh is used to find the neighboring object quadtrees in the direction of the new point and those quadtrees are searched for the new point.

Extracting an image from the quadtree mesh is accomplished by searching for the colors of all pixels in a specified rectangular region. Extracting a coarse image in a quadtree mesh is possible because of the hierarchical nature of each object quadtree.

A Content Addressable Parallel Processor (Verts, Thomson; 1988) can be used to speed up search by assigning one processor to each quadtree mesh node. Each processor would contain the coordinates of the origin and the exponent of two which defines the side length N (which together can be used to determine the bounding box of the object quadtree), and the reference point (point of interest). To search for the color of a pixel all processors would compare, in parallel, the coordinates of the pixel with the bounding box of the quadtree assigned to that processor. Any processors not covering the desired pixel drop out of the search. If there is only one *responder* then the corresponding quadtree is searched for the pixel. If more than one responder remains then several quadtrees overlap the pixel and a conflict may exist.

Resolving Differences in Overlapping Regions

Conflicts arise when two or more object quadtrees differ on the color of a particular pixel. Several techniques can be used to resolve differences. The simplest technique is to always return black for the pixel color. This is in effect taking the logical-OR of the overlapping regions; since at least two differ, one or more must be black. Another method would be to return the value encoded by that object quadtree which is closest to the search pixel (closest according to the mesh reference point).

Although the images are bilevel, the gray (internal) nodes of the quadtree may store an average that is between the black and white values. If so, and if a coarse resolution image is desired, the color of a particular pixel may be computed as the average of all the different values defined by overlapping object quadtrees. Alternatively, the darkest of the competing definitions could be chosen.

Deriving a Quadtree Mesh

Extracting the optimum quadtree mesh from a static image is the subject of ongoing research. Identifying unique, connected objects in the image is relatively simple, but the optimum mesh may involve breaking an object into pieces and assigning an object quadtree to each piece. For example, objects 1 and 2 overlap in the left-hand image of figure 2. An object quadtree applied to the combined figures would require a minimum of 13 leaves (normalized in an 8x8 grid), but object quadtrees for each object require just one leaf apiece.

Building the mesh is much simpler if the objects in an image are known beforehand. For example, constructing a Delaunay Triangulation of a set of N points in the plane has been shown to be of order $O(N \log N)$ (Preparata, Shamos; 1985).

Problems With the Quadtree Mesh Technique

There are several problems with the Quadtree Mesh technique. Conflicting quadtrees have already been addressed.

When extracting the appropriate object quadtree from a static image, it is possible to *over-cluster* or *under-cluster* the image. Over-clustering can occur when there is one object quadtree for each black pixel in the image, regardless of object coherence within the image. Under-clustering can occur when the mesh consists of a single object quadtree, regardless of the complexity or number of objects in the image. The optimum quadtree mesh may indeed turn out to be one object quadtree for each pixel or one object quadtree for the entire image; the process for extracting the object quadtrees and deriving the mesh must be very careful.

If all object quadtrees in a mesh are similar in size, or are of a known maximum size, then it is fairly easy to identify those mesh neighbors that overlap a search pixel. If one object quadtree (on the periphery of the mesh, say) overwhelms the area encompassed by the entire rest of the mesh, then that object quadtree needs to be considered in all search problems, yet its reference point and position in the mesh indicate that it should be rarely involved in neighborhood searches.

EXTENSIONS

Gray Scale Quadtrees

Bilevel images are simple representations of areas: a pixel is either inside an object (black) or outside (white). Realistic images are not simply bilevel but are composed of shades of gray. When constructing the quadtree for a gray scale image it is difficult to determine when the four leaves of a quadrant can be replaced by one larger leaf; it is unlikely that there will be many areas in an image that all have the same gray level. If

exact match is used, all four pixels in an area must have the same gray value to be collapsed into one.

Thresholding is a simple technique to convert a gray region into a black and white region; any pixel above a certain level is changed to black, anything below that level becomes white. This separates figures from their backgrounds in high contrast images, but too much information is lost for this to be useful in general.

In (Gonzalez, Wintz; 1987), a region is considered to be “homogeneous” (all one color) if at least eighty percent of the pixels in that region are within two standard deviations of its average gray level. If the homogeneity constraint is met, the region is assigned the average as its gray value.

An effective way to generate gray quadtrees is to specify an allowable error value such that the pixels in a region are considered to be all one color if no pixel deviates from the average by more than the error number. With an error value of one, for example, collapsing four leaves into one requires that all leaves differ from the average by at most one (plus or minus).

Gray Quadtree Meshes

Meshes can be formed from gray object quadtrees fully as easily as with bilevel object quadtrees. The same techniques apply when resolving overlaps as when dealing with the gray (internal) nodes of bilevel quadtrees. When two or more gray object quadtrees conflict, the color of the search pixel can be determined as either the average of the overlapping values, or as the darkest. Unknown areas (outside all mesh areas) are treated as white or as a “special” value not in the normal image.

An Application of Gray Quadtree Meshes

As a test problem, suppose that the “image” is a digital terrain model, where the “pixel” values represent elevations. The regions encoded by object quadtrees are areas that have known elevations; areas outside all object quadtrees have unknown elevation. The problem is to determine if a line of sight exists between two points in the model. A three dimensional variation of the Bresenham digital stepping algorithm can be used to determine all elevations along the line of sight between the two points. The next step is to examine each point to see if the elevation of the line of sight is above or below the elevation of that point in the model. Each successive point search uses the mesh neighborhood of the previous search. The hierarchical nature of the object quadtrees can be used to quickly derive a coarse model of the terrain, and a measure of the error used in extracting the quadtree from the true image will form a “fuzzy” region within which the true elevation is guaranteed to lie.

Octree Meshes, Gray Octrees and Beyond

The three dimensional analog to the quadtree is the octree. An octree mesh consists of (possibly overlapping) volumes connected together with a three-dimensional meshwork. A meshwork with properties similar to the two dimensional Delaunay Triangulation would define a set of tetrahedra, where any sphere that passes through all four non-coplanar vertices contains no other vertex.

Octrees need not be restricted to bilevel *voxels* (the three dimensional equivalent of a pixel that indicates whether a volume is filled or empty). A *gray octree* can be constructed in the same manner as a gray quadtree. Instead of color, the differing values can represent density, temperature, pressure, or some other multiple-valued phenomenon associated with volumes.

The meshwork, the data structures for decomposing space hierarchically, and the gray extensions to those data structures all have analogs in dimensions higher than three. While such structures are difficult to visualize, the mathematics are consistent.

CONCLUSIONS

The Quadtree Mesh representation has many of the benefits of quadtrees with few of the drawbacks. Multiple, overlapping quadtrees connected together in a meshwork, where each quadtree may be optimized for its subimage, maintain the translation independence and hierarchical representation of each object.

BIBLIOGRAPHY

Chien, C., Agarwal, J. K., 1983. A Normalized Quadtree Representation: Computer Vision, Graphics and Image Processing, Vol. 26 #3 (June 1984), pp. 331-346

Gargantini, I., 1982. An Effective Way to Represent Quadtrees: Communications ACM, Vol. 25 #12 (December 1982), pp.905-910

Gautier, N. K., Iyengar, S. S., Lakhani, N. B., Manohar, M., 1985. Space and Time Efficiency of the Forest of Quadtrees Representation: Image and Vision Computing, Vol. 3 #2 (May 1985), pp. 63-70

Gautier, N. K., Iyengar, S. S., Scott, D. S., 1985. Performance Analysis of TID: IEEE Computer Society Proceedings, Conference on Computer Vision and Pattern Recognition, San Francisco, CA, June 19-23, 1985, pp. 416-418

Gonzalez, R. C., Wintz, P., 1987. Digital Image Processing (second edition), Addison-Wesley, Reading, MA

Jones, L., Iyengar, S. S., 1981. Representation of a Region as a Forest of Quadtrees: IEEE Computer Society Proceedings, Conference on Pattern Recognition and Image Processing, Dallas, TX, August 3-5, 1981, pp. 57-59

Jones, L., Iyengar, S. S., 1983. Virtual Quadtrees: IEEE Computer Society Proceedings, Conference on Computer Vision and Pattern Recognition, Washington, DC, June 19-23, 1983, pp. 133-135

Li, M., Grosky, W. I., Jain, R., 1982. Normalized Quadtrees with Respect to Translations: Computer Graphics and Image Processing, Vol. 20 #1 (September 1982), pp. 72-81

Preparata, F. P., Shamos, M. I., 1985. Computational Geometry, Springer-Verlag, New York

Samet, H., 1983. A Quadtree Medial Axis Transform: Communications ACM, Vol. 26 #9 (September 1983), pp. 680-693

Samet, H., 1984. The Quadtree and Related Hierarchical Structures: ACM Computing Surveys, Vol. 16 #2 (June 1984), pp. 187-260

Samet, H., 1985. Reconstruction of Quadtrees from Quadtree Medial Axis Transforms: Computer Vision, Graphics, and Image Processing, Vol. 29 #3, March 1985, pp 311-328

Scott, D. S., Iyengar, S. S., 1986. TID – A Translation Invariant Data Structure for Storing Images: Communications ACM, Vol. 29 #5 (May 1986), pp. 418-429

Verts, W. T., Thomson, C. L., 1988. Parallel Architectures for Geographic Information Systems: Technical Papers, 1988 ACSM-ASPRS Annual Convention, Vol. 5 (GIS), St. Louis, MO, March 13-18, 1988, pp. 101-107

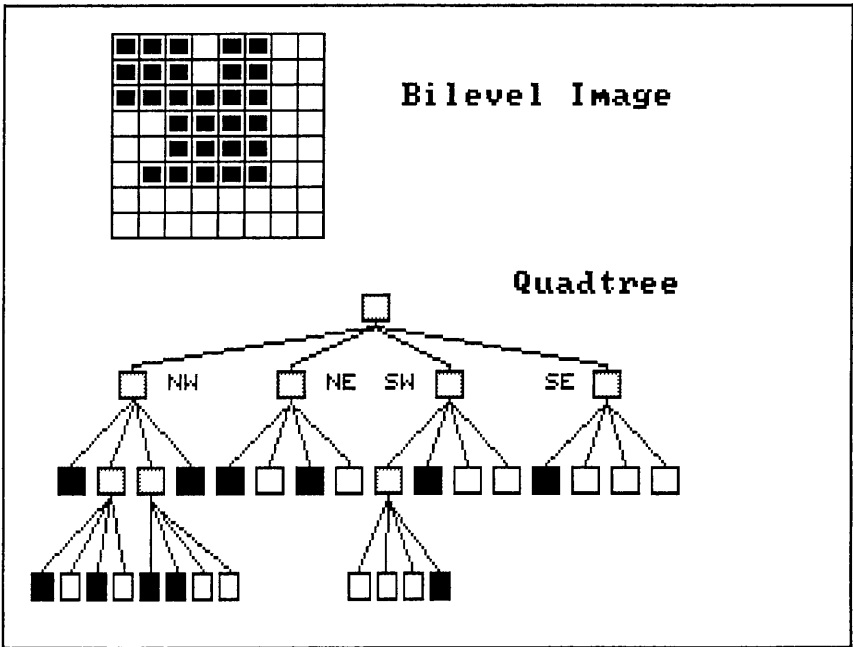


Figure 1: An image and its quadtree.

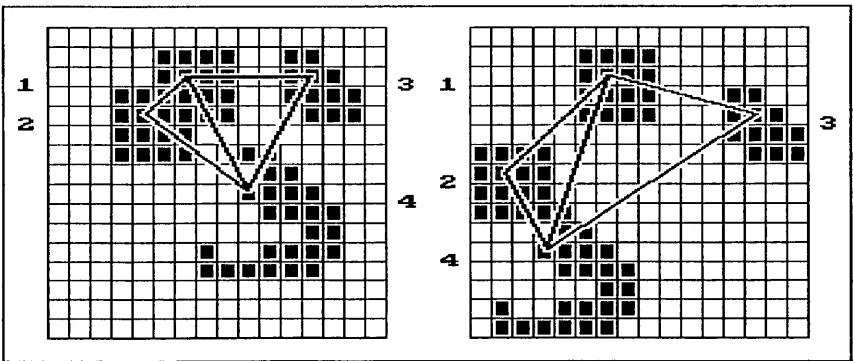


Figure 2: Four objects and their mesh, before and after object motion.

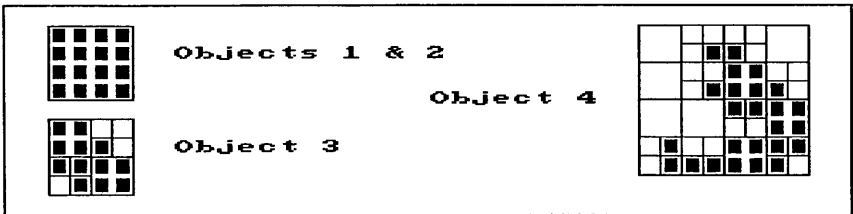


Figure 3: Object Quadtree partitions of the objects in figure 2.

STORAGE METHODS FOR FAST ACCESS TO LARGE CARTOGRAPHIC DATA COLLECTIONS - AN EMPIRICAL STUDY

Andreas Kleiner

Department of Geography, University of Zurich (Irchel)
Winterthurerstrasse 190, CH-8057, Zurich (Switzerland)
e-mail: K247720@CZHRZU1A.bitnet

ABSTRACT

Various cell methods to organize large cartographic data sets under static conditions are implemented and compared. A grid scheme is tested in comparison to methods using a bintree subdivision of space for the organization of background data for drawing base maps. A regular grid index shows best performance and storage efficiency. A pyramid scheme using overlapping grids suited for geographical object storage is compared to the single-level schemes and should be preferred, as it avoids the effort of composing fragmented objects.

1. INTRODUCTION

During the last years, vector-based cartographic data bases were implemented for applications of small and medium size. For practical reasons, large digital cartographic systems have been realized mostly in raster format. At present, efforts on national and international levels are undertaken to collect base map information of whole map series in vector format. Interactive systems require short answer times for spatial queries. Typical systems will display thematic information on top of base maps. While the former will be dynamic information, the latter are large static data collections without frequent change. An important task is to organize these map data for fast queries. The motives for the present work are twofold. First, there exist several individual approaches. It is still a difficult question, which of all proposed methods to use in a real-world application. Comparative tests can hardly be found. Second, all work has dealt with methods for dynamic applications, where data are frequently updated. Static conditions allow other solutions and optimizations of storage arrangement. While an earlier paper (Kleiner and Brassel 1986) presented a theoretical discussion, the subsequent task was to realize an empirical comparative study of different methods. This paper presents the first results.

A detailed discussion of the basic concepts and a survey of the relevant publications can be found in Kleiner and Brassel (1986). Kleiner (1986) is a more complete collection of publications. As the mechanical process of reading from disk is relatively slow, theory tells to concentrate on minimizing this operations for queries. **Cell methods** preserve local proximity of data by attributing one or more storage buckets to a region in geographical space. In dynamic systems, buckets are spread all over the storage by continuous updates. Under static conditions, however, neighbourhood relations should not only be preserved locally by the definition of cells, but also in the arrangement of the buckets on

disk. Sequential reading should be maximized to avoid index accesses and, if physically realized, long mechanical seek time of the reading head. This seek time is critical for sequentially organized optical disks. The one-dimensional order that maximizes preservation of two-dimensional relationships is the **Peano path**, resulting from **bit-interlacing** or **bit-shuffling** of coordinate pairs. The order of the resulting codes corresponds to the traversal of space in a quadtree or bintree.

On the level of individual data, each object should be provided with its enclosing rectangle to speed up clipping at the query window. A basic distinction is made between **background data** - unorganized graphical data for base maps - and **geographical objects** - map information needed for thematic maps, e.g. topological structures of polygon networks for choropleth mapping. In contrast to methods using the flexible bintree scheme also applicable for dynamic data, much simpler static concepts like a regular grid with row and column computation should be considered as well. A **first major question** arises: Is it worthwhile to use a complex, flexible cell method in comparison to the spatially rigid, simple grid scheme? Frank (1983) proposed a scheme with a hierarchy of overlapping cells that allows to always find a cell which an object of arbitrary form and size can fit without the need to cut it at cell borders. The **second major question** is: Is it worthwhile to use a more complex cell scheme for geographic object storage to avoid composing cut parts for each query?

To answer these questions the relevant methods were implemented. In the next sections, the implemented alternatives for storing background data (2.) and those for storing geographical objects (3.) are presented. Each structure and query algorithm is shortly introduced and then the test results are discussed.

2. METHODS FOR STORING BACKGROUND INFORMATION

2.2 Grid Partition

The simplest cell method is to subdivide space into a regular grid. The key of each cell can be computed from its row and column numbers. The spatial query algorithm consists in scanning the cells in the relevant window. The following key organizations are used.

2.2.1 Direct Hashing

The cells are stored row by row and column by column. The addresses of the corresponding buckets can be directly computed, if the buckets are of fixed size. The problem of uneven data distribution leads to empty and overflowing cells. For the latter, overflow chains are formed and appended at the end of the file. Obviously, empty cells have to be stored with one unoccupied bucket.

The size of the cells has to be fixed in advance. If the cells are too large, they are often filled only sparsely; if their size is too small, long overflow chains will appear. A reasonable solution is to strive for cells that fill just one bucket on the average. With inhomogeneous data distributions and a low probability to search in very sparsely occupied regions, it is recommended to use cells that are smaller

than determined by the above given "school-book" rule. This means that the scheme is used as if the average data density was higher.

The advantage of the method is its simplicity, resulting in short central processing time (as well as easy software development and maintenance). No index accesses are needed, but for some cells, whole chains of buckets have to be read, while empty cells are read without retrieving any data. There is no storage overhead for an index structure, but the size of the data themselves is very large in case of inhomogeneous spatial distribution with many empty cells.

2.2.2 Index Hashing

The problem of empty cells can be eliminated by introducing an index corresponding to the spatial grid. Instead of computing a data address, the position in the index is obtained. It contains either a pointer to a data bucket or a nil value. The arrangement is optimized by collecting all buckets belonging to a cell and storing them sequentially. The trade-off between index accesses and accesses to empty cells (and between index storage and storage of empty cells) depends on data characteristics, and the tests cannot lead to a general decision. However, the additional space used for the index should be more than compensated by the saving in data storage.

2.3 Bintree Partition

The bintree is the most flexible regular tree division of space of "trie" type. In this application field, it has to be preferred to the less economical quadtree, as with a quadtree split always two bintree splits are performed at once. This scheme is combined with index methods using coordinate shuffling.

The question at hand is how to organize the spatial query. An elegant algorithm was published by Orenstein and Merrett (1984) and is the basis for the present implementation. The idea is to utilize the hierarchy of the space partition for a recursive algorithm. A bintree representation of the search window is constructed to be filled with the largest possible cells (getting smaller towards the borders). These cells then are subject to searches. The algorithm is limited by the depth of the real tree. For each search cell involved, beginning at the root of the tree, one of three cases occurs:

1. The cell lies completely outside the region of interest. It is neither searched nor further split.
2. The cell lies completely inside the region of interest. All data cells inside the search cell are read and processed.
3. The cell overlaps the region of interest. It is split, and the two sub-cells are handled recursively.

The search cells are processed in Peano order and within each search cell, all potential sub-cells form an uninterrupted segment of the Peano path. The search of a cell always begins with an index access to get the first data address. The rest can be read sequentially without the use of the index. The formation of the largest possible search cells guarantees maximum sequential reading.

A data cell can be smaller than, equal to or larger than the search cell. In the latter case, it must be ensured that the data cell is not read more than once.

Without going into details, it shall be mentioned that further improvements are combined with the elimination of recursion. Splitting of a cell or forming a parent cell involves only one bit of its Peano key.

2.3.1 EXCELL

This method by Tamminen (1983) is based on Extendible Hashing, which uses a directory of data addresses for each potential hash value. Adaptation to the data distribution is achieved by assigning of one or more pointers to the same data bucket based on the trie principle. If coordinate shuffling is used as hash function into the index, the assignment of the index to the data is an implicit bintree. In this static implementation, empty directory positions are filled by the address of the next non-empty cell.

The index of very large data sets cannot be held internally and must be kept on disk, distributed to pages. The effort to find the first data address involves the computing of the index position and at most a single access to the page containing the relevant part of the index. The size of the index may be a problem. As it exists in the maximum resolution of the key space, large cells occupy large parts of the index with the same address.

2.3.2 Two-level EXCELL

Large Extendible Hashing indices must be stored externally, particularly if there are many data layers with their own indices. However, such a structure should take advantage of the growing central storage capacity of modern hardware. For this purpose, an extension of EXCELL was developed, where a two-level index is used: an on-line index of fixed depth loaded into memory (from now on called internal index) and external indices for regions of higher resolution. In the practical implementation the external indices are kept in one file. As these local refinements to the first-level resolution may have any depth, a second internal structure is needed parallel to the first, i.e. a directory that indicates the corresponding "sub-depths". Searching is performed by computing the position in the internal index. This index item either points to a data bucket or to an external subdirectory. In the latter case, the pointer marks the beginning position of the relevant index in the external file. The external index now contains the required data address.

As the size of the index grows by the power of two, the overall index size decreases strongly with the introduction of an internal index. The effort of a search is reduced, when a data bucket is directly addressed by the internal index, as there is no need for a disk access; it is slightly worsened, when an external index position has to be accessed by two subsequent hash function computations. A similar approach was proposed by Davis (1986) for dynamic applications with a combination of EXCELL and interpolation-based index maintenance.

2.3.3 Bintree with B-tree

B-trees as the "classical" index structure in conventional data bases, can also be used to index bintree cells. The B+-tree is the variant used for range search, where the leaves form a sequential list of all existing keys (Comer 1979, Abel

1984). As the static conditions allow sequential searching without the use of an index, the original B-tree is better suited. The worst case effort to find the first data address with a B-tree is a logarithmical function of the size of the data. Average behaviour is influenced by buffering.

2.4 General Remarks on the Test Environment

The results of the specific environment will be discussed in respect to their general application. The present implementations are written in Pascal and run on an Apple Macintosh II with a Winchester disk of 30 ms average access time. The size of index pages was chosen as one disk block (512 bytes), data buckets consist of two blocks. Thus all figures for data file accesses have to be divided by two to get the actual number of data buckets. The measurements are comparable among the methods, but specific to hardware parameters, i.e. type of disk and the power of the CPU. An important distinction has to be made between **disks for direct access** and **sequentially organized disks**. The average access time of the optical CD-ROMS, which belong to the latter kind, is more than ten times slower than that of magnetic disks.

2.5 Test Results

The data set used for the present tests is part of the World Data Bank II and contains line data. The sizes of the different index and data files are indicated in table 1. WDB stands for the original World Data Bank files. With grid storage (G) that includes empty cells, the required file space is more than doubled. Using the grid structure with an index (G/I) to avoid storing of empty cells, the size of data is reasonable, while the index is very small. Attention should be paid to the large size of the bintree structure. In fact, the bintree with its spatial flexibility has a low bucket occupancy. **"Vertical" adaptation to variable data density by allocation of buckets assigned to one cell in a chain obviously is more economical than the "horizontal" adaptation of splitting cells in geographical space.** In order to explain the higher storage efficiency of the grid scheme, the consequences of a bucket overflow have to be analyzed. If an additional bucket is appended to a grid cell, only one line is affected that will continue in the new bucket. If a bintree cell is split, all contained lines can happen to be geometrically cut by the new borderline; even the same line may multiply across the border. Each intersection forces the introduction of new boundary points and the creation of additional enclosing rectangles.

The EXCELL directory (E/I) is almost as large as the data themselves. The use of the two-level EXCELL (E/2) leads to a significant reduction of the file of external directories, although the present test implementation used an internal directory of only depth 7. The B-tree (B/B) has a depth of 4.

Queries were carried out with variations of spatial range and in regions of different data density. Table 2 shows typical results for different sizes of the query window in areas of reasonable data density. The examples are chosen to be representative within a certain variance. The answer to the basic question is clear with respect to the present environment: **The grid solution performs significantly better than the bintree methods.** In the relation of internal processing to external reading the latter can even be neglected. Looking at

external processing, the results differ according to the size of the required map window. In general, the bintree partition is thrifter with accesses only for small windows, whereas large regions need fewer accesses to data buckets with the grid organization. How is this phenomenon to be explained?

In view of the **number of data accesses**, two factors have to be considered: the **adaptability to data distribution in space** and the **bucket occupancy**. The first factor is of significance only at the borders of the query window. If an area of high data density lies at a border of a "horizontal" partition in the geographical domain, only small cells hardly overlapping the boundary are retrieved. With large grid cells and "vertical" overflow chaining, a whole chain is read whose contents only partially overlap the window. In the internal regions, however, the second factor is decisive. The fact that the grid is more economical with respect to storage utilization results in a lower number of buckets. As a result, the border conditions dominate in smaller query regions, whereas bucket occupancy is crucial for large regions. The balance of the two effects depends on the bucket size. Small buckets have a better adaptability. On the other hand the relation of data to header information is worse, mainly due to higher object fragmentation. It shall be emphasized that the number of disk accesses would be crucial only if a processor about thirty times as fast was used.

The number of index accesses is not of significant influence, but it may be noted that the indexed grid method needs less accesses than EXCELL. Supposing the same cell partition of space, ordering the cells according to the Peano path should allow more uninterrupted sequences that can be read in sequential order. Why did the bintree scheme lead to more interruptions with need of index accesses? The average cell size of the grid implementation with the allocation of possibly several buckets is much larger than that of the bintree solution.

Among the grid solutions the index method with index can be preferred, since the large data file is substantially reduced and the influence of the index query on time is very small. If - in contrast to the examples used - searches are done in sparsely occupied areas, the indexed solution eliminating empty cells is even better.

The results of single-level and double-level EXCELL are almost equal. The internal index depth of the latter was chosen small, and it should give the same effect as a very large data set with a deep first-level index. However, in the present examples mostly external accesses occurred, together with many changes between different external indices. This should not be expected with larger files and greater internal depth, and the two-level variant can be expected to be superior in general.

The use of the B-tree instead of external hashing seems to be a valid alternative in the present implementation, but this result cannot be generalized to data sets of arbitrary size because the tree and the search time grow with the size of data.

3. METHODS FOR STORING GEOGRAPHIC OBJECTS

3.1 Organizations for Background Information

Point objects are stored in one of the above organizations. The common solution for objects of spatial extension (lines, areas) is to use these same methods, but to cut each object at cell boundaries. Queries thus require the extra effort of composing all object parts.

3.2 Overlapping Pyramid Hashing

The Field Tree was proposed by Frank (1983) to overcome the problem of cutting objects. It is a hierarchical structure, where each object is stored in a cell large enough to accept it in its integrity. The size of the cells is not determined by the spatial distribution of data, but by the size of its objects. Thus for the same region, cells of different levels of the tree can exist. The Field Tree is similar to a quadtree. Because in a quadtree partition objects cutting a main cell boundary can only be stored in the root, each level of the tree is displaced by half a cell. Small objects crossing high level borders now fit lower level cells because the cells overlap.

Similar to the background data organizations, the simple grid concept should be tried for the Field Tree concept, leading to a hierarchy of grids. It is implemented as a pyramid of displaced grid indices, allowing access by hashing instead of a tree traversal. Each level corresponds to the structure described earlier and is accessed by row/column hashing, however with different origins on each level. As the number of empty cells is very large and it is important to store only non-empty cells, the grid solution with index is used. The high-level indices may be kept in memory. Each query is performed independently on all levels. An overlapping pyramid applied to image representation is mentioned by Samet (1984).

The structure is not very flexible with respect to data distribution, and typical geographical objects are not very small. Therefore, variable size buckets (in multiples of a block) are used in the implementation. One main restriction to the use of this scheme is its limitation to object types of relatively small size. For example, it would be nonsense to store the coastlines of the oceans in such a structure, with the whole Pacific in one gigantic cell, and then ask for the coasts of Hawaii in a query.

3.3 Test Results

The test example uses administrative boundaries of Switzerland as line objects (arcs). Table 3 shows a typical result. The effort to search for identical beginning and end points of cut objects is far greater than the time for the query of the initial data. With the simple concept of different grids, the problem of the complex tree of cells is eliminated and overall performance is clearly better than with the methods using one level of cells. The search time is even shorter than that of the bintree methods. The test shows that **avoiding the composition of object parts is worthwhile when using the fast pyramid hashing**. Only with a substantially different relation between processor and disk speed, the low load

factor of the overlapping pyramid buckets could be of significance.

4. CONCLUSIONS

The use of cell methods to organize large spatial data sets is important, but by concentrating on external read operations, the internal processing effort must not be neglected. The results show that grid hashing performs faster than bintree methods with Peano key indexing due to its simple internal processing. Even with a combination of a very fast processor and slow optical disks leading to a crucial influence of disk accesses, bintree methods are superior only with small query windows. These results are valid for line and area data in vector format, where bucket occupancy turns out to be more economical with overflow chaining than with geometrical splitting of cells. For reasons of storage efficiency the grid method with index should be preferred. The use of the overlapping pyramid scheme to avoid cutting objects along cell borders is worthwhile unless strong processing power would reduce the composing effort drastically.

ACKNOWLEDGEMENTS

Prof. Kurt E. Brassel has kindly reviewed the text. This contribution is gratefully acknowledged.

REFERENCES

- Abel, D. J. (1984): "A B+-Tree Structure for Large Quadtrees", *Computer Vision, Graphics, and Image Processing*, 27, pp. 19-31.
- Comer, D. (1979): "The Ubiquitous B-Tree", *Computing Surveys*, Vol. 11, pp. 121-294.
- Davis, W. A. (1986): "Hybrid Use of Hashing Techniques for Spatial Data", *Proceedings Auto Carto London*, Vol. 1, pp. 127-135.
- Frank, A. (1983): "Probleme der Realisierung von Landinformationssystemen, 2. Teil: Storage Methods for Space Related Data: The FIELD TREE", *Institut für Geodäsie und Photogrammetrie, Bericht*, Nr. 71, Zurich: ETH, 63 pp.
- Kleiner, A. and K. E. Brassel (1986): "Hierarchical Grid Structures for Static Geographic Data Bases", *Proceedings Auto Carto London*, Vol. 1, pp. 485-496.
- Kleiner, A. (1986): "Data Structures for Spatial Data Bases", in: R. Sieber and K. E. Brassel (ed.): *A Selected Bibliography on Spatial Data Handling: Data Structures, Generalization and Three-Dimensional Mapping*, Zurich: University of Zurich, pp. 3-19.
- Orenstein, J. A. and T. H. Merrett (1984): "A Class of Data Structures for Associative Searching", *Proceedings of the Third ACM SIGACT-SIGMOD Symposium on Principles of Database Systems*, pp. 181-190.
- Samet, H. (1984): "The Quadtree and Related Hierarchical Data Structures", *ACM Computing Surveys*, Vol. 16 (2, June).
- Tamminen, M. (1981): "The EXCELL Method for Efficient Geometric Access to Data", *Acta Polytechnica Scandinavica, Mathematics and Computer Science Series*, No. 34, Helsinki: Univ. of Technology, 57 pp.

Table 1: File size with different methods

	WDB	G	G/I	E/1	E/2	B/B
index size (K)	81	-	27	4097	540	179
data size (K)	3173	8913	3612	5357	5357	5357

Table 2: Query of line data with different methods

	G	G/I	E/1	E/2	B/B
query window: 0.5° x 0.5°					
search time (s)	1.03	0.99	6.05	6.53	6.57
internal portion (s)	0.11	0.12	5.43	5.91	5.91
external portion (s)	0.92	0.87	0.62	0.62	0.66
disk accesses	42	44	26	26	28
index accesses	-	2	2	2	4
data accesses	42	42	24	24	24
bucket occupancy (%)	45	45	31	31	31
used data (%)	25	25	76	76	76
query window: 1.0° x 1.0°					
search time (s)	1.59	1.46	10.33	11.14	11.24
internal portion (s)	0.13	1.12	8.96	9.78	9.77
external portion (s)	0.46	0.34	1.37	1.36	1.47
disk accesses	70	72	70	66	69
index accesses	-	2	6	2	5
data accesses	70	70	64	64	64
bucket occupancy (%)	45	45	32	32	32
used data (%)	41	41	76	76	76
query window: 2.0° x 2.0°					
search time (s)	4.16	3.63	16.58	17.76	18.03
internal portion (s)	0.19	0.17	12.35	13.47	13.44
external portion (s)	3.97	3.46	4.23	4.29	4.59
disk accesses	172	174	213	213	212
index accesses	-	4	13	13	12
data accesses	172	170	200	200	200
bucket occupancy (%)	43	44	32	32	32
used data (%)	66	66	85	85	85
query window: 4.0° x 4.0°					
search time (s)	7.44	6.66	20.40	21.44	21.74
internal portion (s)	0.21	0.23	12.30	13.46	13.45
external portion (s)	7.23	6.43	8.10	7.98	8.29
disk accesses	318	318	427	424	425
index accesses	-	6	15	12	13
data accesses	318	312	412	412	412
bucket occupancy (%)	41	42	32	32	32
used data (%)	88	88	94	94	94

Table 3: Query of line objects with different methods

query window: 130 x 130 km	G	E	OP
search time (s)	1.22	4.87	2.42
object processing time (s)	33.83	40.55	8.59
total time (s)	35.05	45.42	11.01
disk accesses	68	88	92
index accesses	-	8	-
data accesses	68	80	92

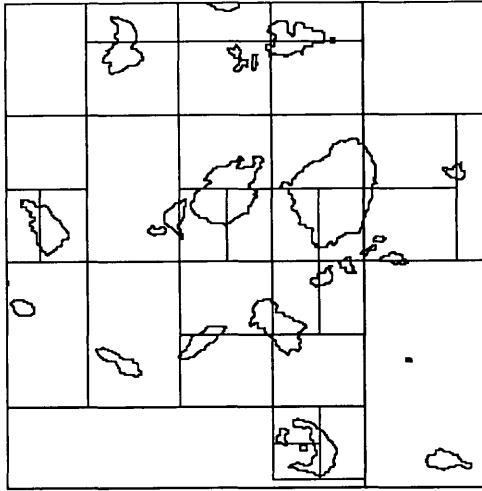


Figure 1: Greek islands organized in a bintree

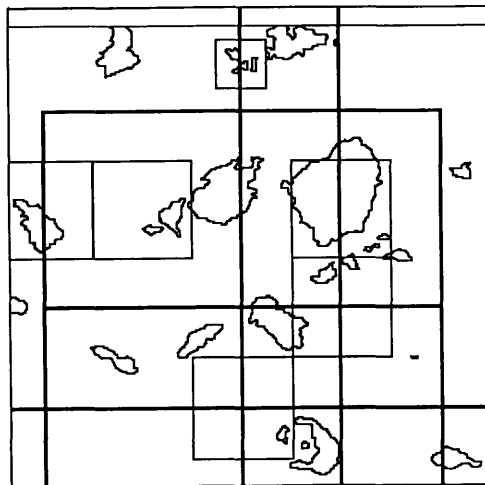


Figure 2: Greek islands organized in a pyramid of overlapped cells

SOLVING SPATIAL QUERIES

BY RELATIONAL ALGEBRA

Robert LAURINI, Françoise MILLERET
Laboratoire Informatique Appliquée
Institut National des Sciences Appliquées de LYON
69621 VILLEURBANNE Cedex, FRANCE

Abstract :

In conventional Geographic Information Systems, computational geometry is used to solve spatial queries such as point-in-polygon, region and vacant place queries. In this paper, we propose a formalism (Peano relations) based on linear quadtrees and Peano space-filling curves which allows the solving of the previous queries by a tuple algebra. In essence, it is a relational algebra taking into account the extensional/intensional approach of spatial data. Some examples are taken from urban planning and we conclude this paper by emphasizing several aspects of geomatic reasoning.

Keywords :

GIS, spatial data modeling, quadtree, Peano keys, algebra, spatial query, geomatics, spatial reasoning.

I - INTRODUCTION

Spatial database management systems address to application dealing with geometric and topological data especially in geomatics. One important issue in their design is how to handle queries against geometric information. Some off-the-shelf systems propose two kinds of data:

- attribute data to which relational algebra can be applied,
- and geometric or graphic data for which computational geometry is used to solve spatial queries.

However, when one has to solve a query combining criteria with alpha numeric data and geometric data, he has to make a mixture of relational algebra and computational geometry.

Due to Peano relations, we will show that spatial queries can also be solved by a tuple relational algebra. So, the goal of this paper will be to present a new methodology for answering queries.

Let us examine a small example. Should we ask to retrieve the number of trees in a zone or in a lot of zones, no problem will arise and the answer is obtained by ordinary relational algebra. But if we are interested in the number of trees in a region defined by its boundary, we need to use computational geometry (see Figure 1). In this paper, we will show that Peano relations can allow the solving of this query by relational algebra.

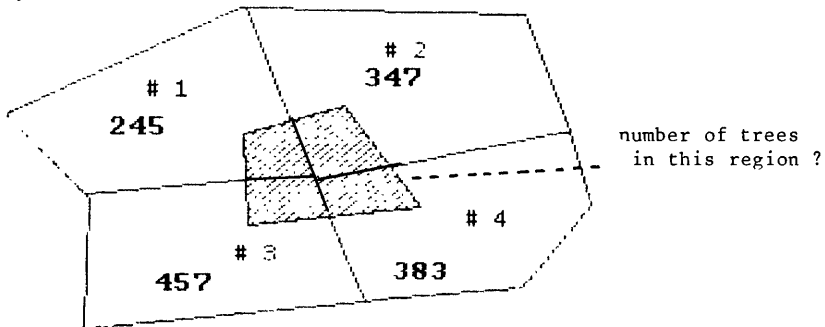


Figure 1 : Example of query not solvable by relational algebra :
what is the number of trees in the hatched region

In this paper, we will first present the Peano tuple algebra, then the typology of spatial queries in order to answer a multimedia spatial query example. And we will conclude by some aspects of geomatic reasoning.

II - PEANO TUPLE ALGEBRA

Let us first present the Peano Relations model and second its algebra.

2.1 Peano relations

In several papers (LAURINI, 85, 87 and LAURINI-MILLERET, 87) we have defined a spatial database model whose characteristics are :

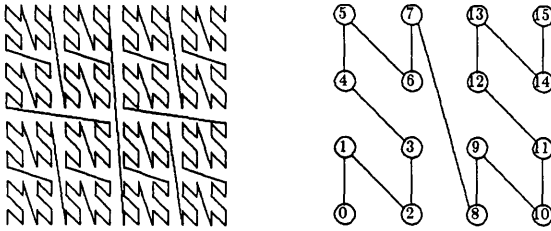


Figure 2 : Excerpts of the Space-filling Peano N-curve

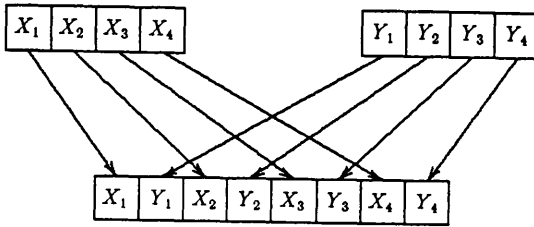


Figure 3 : Obtaining Peano keys by bit interleaving

Peano Key	size	color
0	2	black
4	1	black
5	1	white
6	1	white
7	1	white
8	1	white
9	1	white
10	1	black
11	1	white
12	2	white

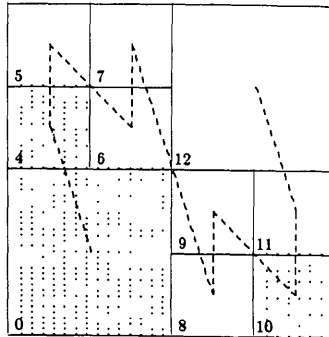


Figure 4 : Examples of a quadtree and its description by Peano relations

- area orientation avoiding the infinite number of tuples
- based on Peano space-filling curves
- based on linear quadtrees and octrees
- use of tuple algebra to solve spatial queries.

In this model, a squared 2D space is described by a recursive splitting into homogenous quadrants (Quadtree), (See SAMET, 1986) and these quadrants are sorted by their Peano Key. In the 3D space we deal with octants. Peano keys p derives from fractal space-filling curves (Figure 2) and the more practical way to obtain them is by the bit interleaving of the x and y coordinates (Figure 3). Peano key based quadtrees are also named linear quadtrees (GARGANTINI, 1983) and Peano keys are also called Morton sequence (MORTON, 1966), z -value by ORENSTEIN (1986) and tesseract arithmetic by DIAZ-BELL (1986)

Let note a Peano relation : $R(p, a, A)$

in which

- p stands for a Peano key
- a the size of the square/cube
- A a set of domain attributes.

An example of an object described by a Peano relation is given in Figure 9. Often, to shorten, we can exclude white or void squares giving :

$R(\# \text{ object}, p, a)$

Bearing in mind that a tuple describes a square/cube, it is easy to see that it can be split into 4 (respectively 8) other tuples. So it is an intensional/extensional way of describing space and the rule is

"One tuple can always be split into 4 (8) tuples"

In order to deal with consistent and compact objects, 3 conformance levels are necessary:

- well positionned squares/cubes
- overlapping elimination
- maximum compaction.

2.2 Manipulation

In (LAURINI, 1987) the Peano tuple algebra for manipulating object is given. Beside geometric and boolean operations, relational operations are very useful for manipulations. For instance the Peano join can be used to solve point-in-polygon and regions queries.

Let us consider an example. A scene consisting of three objects, A , B and C is described by means of a Peano relation SCENE ($p, \# \text{ object}, a$) and we want to test a region REGION (p, a) in order to know what are the objects within it. See Figure 5. The result is given, first by a Peano join between SCENE and REGION and second by a projection of this result. In the example, the tuple SCENE ($B, 52, 2$) can be disaggregated into four tuples SCENE ($B, 52, 1$), SCENE ($B, 53, 1$), SCENE ($B, 54, 1$) and SCENE ($B, 55, 1$).

Scene		
object	Peanokey	size
A	16	2
A	24	2
A	28	1
A	30	1
B	29	1
B	31	1
B	48	2
B	52	2
C	0	4

Region	
Peanokey	size
30	1
31	1
52	1
53	1
55	1
62	1

Result of the join		
object	Peanokey	size
A	30	1
B	31	1
B	52	1
B	53	1
B	55	1

Project result
object
A
B

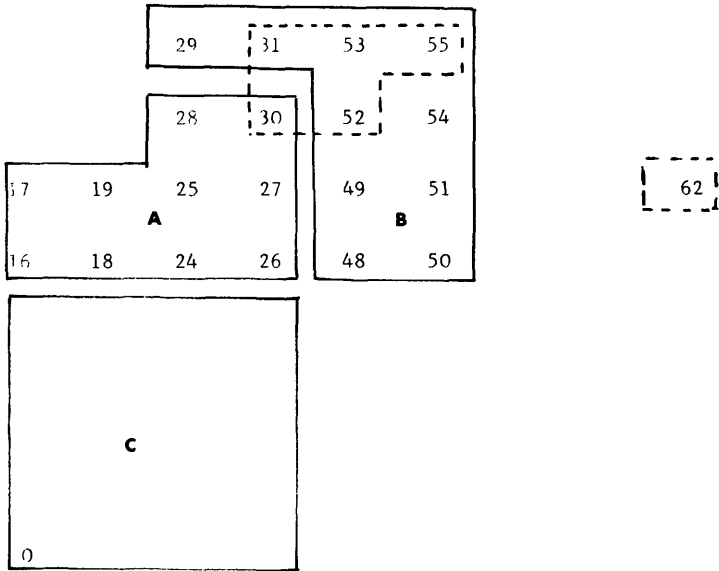


Figure 5 : Examples of spatial object (A,B) and a region query

III - SOLVING SPATIAL QUERIES

The role of spatial queries is to provide algorithms to solve questions. See LAURINI-MILLERET (1988). Among them, the more important seem to be :

- point-in-polygon query,
- region query,
- vacant places.

We have recently established that Peano relations give faster algorithms to solve spatial queries than wireframe representation.

3.1 Point-in-polygon (Fig. 6a)

Starting from a x_0, y_0 point, the problem is to find what objects it belongs to. An example is to determine who is the landowner of a point in a cadaster. Let us suppose we have n plots of land.

With a wireframe representation, the solution is given by the half-line algorithm whose complexity is $O(n)$ (PREPARATA-SHAMOS, 1986). With the cell-oriented representation based on Peano relations, the complexity becomes $O(\log n)$ (LAURINI, 1987).

3.2 Region query (Fig. 6b)

Here, starting from a zone called region, one has to determine what are the objects belonging to it. It is the same problem as the point-in-polygon query except that the point is replaced by a zone. As an example in town planning, we can try to retrieve the landowners affected by the creation of a new freeway, or the list of urban objects in a zone defined by its boundary.

With the wireframe representation, one has to perform an algorithm based on an intersection which is very complex to design. With Peano relations, this query is solved by a Peano join.

3.3 Vacant places (Fig. 6c)

Here, the problem is to retrieve vacant places within a predefined zone. Wireframe representation leads to a geometric difference algorithm more difficult to be written than a relational difference algorithm with Peano relations.

In a same way, we have shown that Peano relations allow the easy solving of spatial query by means of a join operator taking into account the extensional/intensional aspects (Peano join). See LAURINI (1987).

3.4 Other spatial queries

Among other spatial queries, let us mention distance query. For instance, we want to retrieve all parcels within a distance of 3 km from a precise one.

With the wireframe representation, in the first step, one has to determine a region built from the given parcel and the distance, and to apply the region query. The computation of the shape of this region is not very simple to perform, especially in the case of holes and concavities.

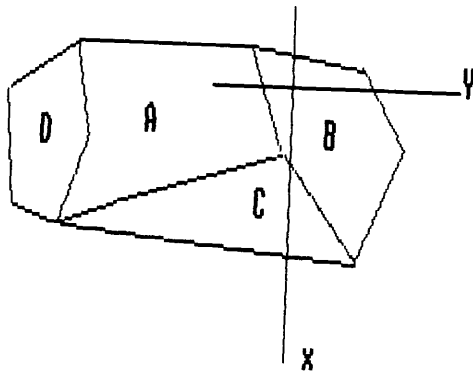
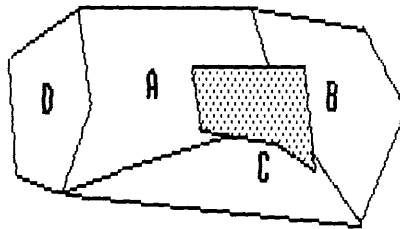
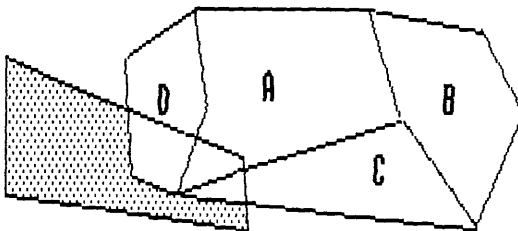


Fig. 6a: Point-in-polygon query



Answer: A, B and C

Fig. 6b: Region query



Answer: Hatched portion

Fig. 6c: Vacant place query

Figure 6: Examples of spatial queries

However, with quadtrees, we can determine the region by constructing the union of all quadrants and their neighbors within the distance. After this step, the result is given by a Peano join.

3.5 Example of a multimedia spatial query

Suppose, after a flooding, we want to retrieve all farmers affected by this flooding in order to indemnify them. For that, let us start from the cadaster and aerial photographs of flooded fields. After having given the structure of the land data, and pixel-based photographs, we present the solving process.

a) Land data structure

Let us have a relation giving a parcel and its landowner and three other relations for parcel boundary description :

- R_1 (# Parcel, # Farmer)
- R_2 (# Parcel, # Segment)
- R_3 (# Segment, # Point 1, # Point 2)
- R_4 (# Point, x, y)

b) Aerial photographs

Suppose an aircraft has taken digital photographs of flooded area with several gray levels. Moreover, suppose the exact position of each photo in term of coordinates and orientation is known.

- P_1 (# Image, x pixel, y pixel, gray level)
- P_2 (# Image, x pos, y pos, length, width, orientation)

c) Query solving (Figure 13)

To solve this spatial query implying geometric objects described with various geometric representation, its seems interesting to map into the linear quadtree representation (Peano relations) whose main advantage is to use relational algebra to solve spatial query. See LAURINI 1987, or LAURINI-MILLERET 1987 and 1988 for more details. First, let us deal with aerial photos for which a relational restriction must be applied to cancel all pixels not corresponding to the water. By examining gray levels of pixels, this operation will be performed so giving A_1 corresponding to P_1 reduced to water.

- A_1 (# Image, x pixel, y pixel)

Second, all these pixels have to be positioned in the coordinate system and geometrically corrected due to photographic distortions transforming some pixels into rectangles :

- A_2 (# Image, x corrected, y corrected, length, width)

The next step will be to regroup all these relations in order to cover the whole territory by quadtrees governed by Peano keys :

- FLOODING (Peano Key, size).

In the same manner, we have to transform land information into Peano relations : so, starting from R_2 , R_3 and R_4 , we can construct the following relation :

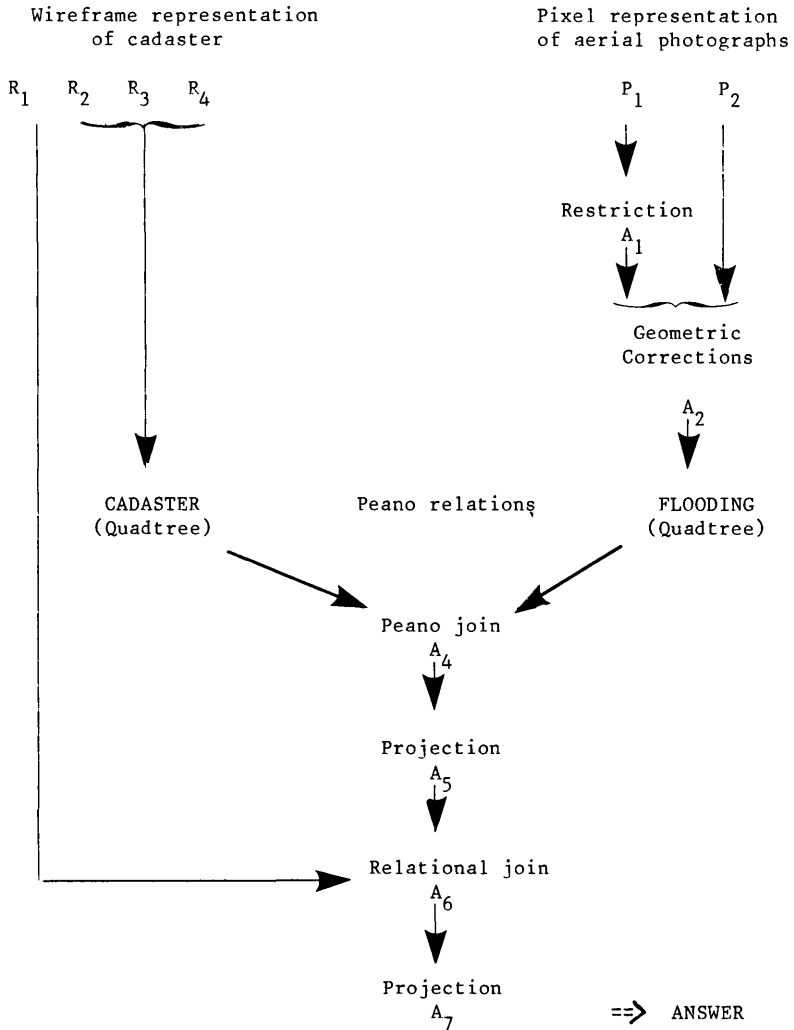


Figure 13 : Solving algorithm for the multimedia query

CADASTER (# Parcel, Peano Key, Size)

To obtain the result, a Peano join has to be performed between FLOODING and CADASTER to give A_4 corresponding to only flooded plots :

A_4 (# Parcel, Peano Key, Size)

Now, to know the list of flooded farmers, in a first step, we will perform a projection on A_4 to give only the parcels (A_5) ; then a relational join of A_5 with R_1 to give A_6 which will be followed by a projection to get A_7 which is the answer.

- A₅ (# Parcel)
- A₆ (# Parcel, # Farmer)
- A₇ (# Farmer)

So, the result is obtained through an amalgamation of computational geometry and relational and Peano algebras.

IV - CONCLUSIONS

The scope of this paper was to show that some spatial queries can be easily answered by tuple algebra when solid objects are described by Peano relation. A multimedia example, taken from urban planning has illustrated this fact.

For the design of a spatial DBMS, we do think that the representation must be chosen not only from storage criteria but also from the facility to solve spatial queries. Peano relation and cell enumeration methods are very good candidates for the foreground of this design.

BIBLIOGRAPHY

DIAZ BM, BELL SBM (1986) Spatial data processing using Tesseral Methods. Publ. by Natural Envir. Research Council (UK), Sept 1986.

GARGANTINI I (1983) Translation, Rotation and Superposition of Linear Quadrees. International Journal of Man-Machine Studies. Vol 18, 3, March 1983, pp 253-263.

LAURINI R (1985) Graphics Data Bases Built on Peano Space-Filling Curves. EUROGRAPHICS'85, Nice, Sept.8-13/1985. pp 327-338. Ed. by C.E. VANDONI, NHPC.

LAURINI R (1987) Manipulation of Spatial Objects with a Peano Tuple Algebra. University of Maryland CfAR, Technical Report. CAR TR 311.

LAURINI R, MILLERET F (1987) Peano Relations in CAD/CAM Databases. Int. Conf. IEEE "Data and Knowledge Systems for Manufacturing and Engineering", Hartford, Connecticut October 19-20/1987.

LAURINI R, MILLERET F (1988) Spatial data base queries: relational algebra versus computational geometry. "IVth International Conference on Statistical Scientific Data Base Management", Rome, Italy, 21-23 June 1988, Vol 2, pp 23-44. Edited By M. RAFANELLI.

MORTON GM (1966) A Computer Oriented Geodetic Database and a New Technique in File Sequencing. IBM Canada-Ontario report, March 1966.

ORENSTEIN J (1986) Spatial Query Processing in an Object-Oriented Database System. Proceedings of ACM/SIGMOD'86, Washington DC. pp 326-336. Edited by C. ZANILOLO.

PREPARATA F, SHAMOS M (1986) Computational Geometry: an Introduction. Springer-Verlag.

SAMET H (1984) The Quadtree and Related Hierarchical Data Structures. Computing Surveys. Vol 16, June 1984, pp 187-260.

SPECULATIONS ON SEAMLESS, SCALELESS CARTOGRAPHIC DATA BASES

Stephen C. Guptill
U.S. Geological Survey
521 National Center
Reston, Virginia 22092

ABSTRACT

The idea of a seamless, scaleless data base of digital spatial data for use in automated cartography or geographic information systems has intrigued researchers for a number of years. Yet reviewing the plans of mapping agencies in the United States and Europe for their digital data bases in the 1990's shows partitioning by scale, space, and time. Why is this so? Is a seamless, scaleless data base still a quixotic quest, or have the conceptual models of spatial data and computer technology advanced to the point that such a goal is achievable? This paper contends that the impediments are now pragmatic concerns and not technological ones.

INTRODUCTION

The first uses of digital cartographic data were primarily geared toward the automation of the traditional map drafting process (Tomlinson, 1988; Rhind, 1988). This concept quickly expanded to the notion of interactive roaming through large cartographic data bases unhindered by map sheet boundaries or scale of presentation (Radlinski, 1974; Fields, 1978). Trailing these ideas was the actual creation of digital cartographic data bases and computer software to manipulate the data.

The first systematic collection of digital cartographic data sets began in the 1960's with the Canada Geographic Information System and was followed by efforts in the United Kingdom at the Experimental Cartography Unit, the Ordnance Survey, and the Military Survey, and in the United States by the Central Intelligence Agency, the Defense Mapping Agency, the Bureau of the Census, and the Geological Survey. Data manipulation software was largely written by the data-producing agency or academia, although some commercial software was available in the late 1960's and early 1970's.

Over the subsequent two decades, digital cartographic data has become much more prevalent, and capabilities of geographic information system (GIS) software to *manipulate these data have increased dramatically*. However, the concept of browsing through a seamless, scaleless data base has yet to be realized.

A seamless data base implies an ability to query, display, retrieve, or otherwise traverse the contents of a large spatial data base without limitations imposed by the spatial extent of the data. For example, a command to display the Mississippi River would yield the entire river, not just a portion of it. A scaleless data base implies an ability to transition from one level of detail to another appropriate to the scale of the display or precision of the data analysis. For example, select a feature, say Dulles Airport, and display its location with a point symbol at a scale of 1:2,000,000, zoom in on the airport, with runways appearing at a display scale of about 1:100,000, then, as the display scale increases, more detail, such as buildings, fuel tanks, and parking lots will appear.

Achieving these capabilities requires advances on two fronts: (1) availability of spatial data amenable to this environment, and (2) capacity of spatial data base management systems (DBMS) or GIS's to manipulate the data. Data producers must create data that can be easily placed into a seamless, scaleless data base environment. Data base system developers must then provide the tools necessary for handling the data.

DATA PRODUCERS

As noted above, national mapping agencies, including the U.S. Geological Survey have been producing digital cartographic data for over a decade. The data structures and underlying data models (some utilizing topology) used in collecting this information, for the most part, have not changed since they were developed in the late 1960's or early 1970's. However, during this time, the tasks for which the data were used became increasingly sophisticated, placing information demands on the data that were not planned for in their initial design.

In response, over the last several years, various mapping agencies have been designing data models for use in their current or future information systems. Much of this work is still ongoing and few references to the published literature are available. In those cases without published references, either agency representatives or internal reports are cited. The trend among these agencies is a data model built on a basic set of topological elements and superimposed with a set of cartographic features. This type of design may ease the construction of scaleless data bases. What is less clear is that national mapping agencies are committed to building seamless, scaleless data bases.

U.S. Geological Survey

In the case of the U.S. Geological Survey, one of the sources of the demands for change arises from the Survey's National Mapping Division, which is undertaking a major system development activity called Mark II. Mark II will be a digital cartographic production system with the National Digital Cartographic Data Base (NDCDB) at its hub. Information in the data base will reflect the data content of the National Mapping Program's standard map series. This information will be periodically revised and new graphic products generated using computer-assisted cartographic methods. Maintaining the information required to support these processes is a driving force behind the design of a more comprehensive data model (Morrison and others, 1987). Additionally, the growing sophistication of GIS's and the increasing diversity of applications involving GIS technology and spatial data are beginning to demand a more flexible and comprehensive model for spatial

information. This linkage between advancing GIS capabilities and the need for more advanced data structures has been explored by Goodchild (1987).

Responding to these demands, the Geological Survey (as both a data supplier and data user) has begun the design of an enhanced version of the digital line graph, termed Digital Line Graph - Enhanced (DLG-E). In simple terms, the DLG-E begins with the topological model used in the Survey's present DLG format (U.S. Geological Survey, 1986) and builds a cartographic feature layer upon the topology. The feature definition is open-ended, allowing users to define additional features of interest. Cartographic entities will be described using objects, attributes, and relationships. Additionally, recommendations contained in the "Proposed Standard for Digital Cartographic Data" (*The American Cartographer*, 1988) regarding data quality information and formatting will be followed. Details on the DLG-E design are given "Designing an Enhanced Digital Line Graph" (Guptill, and others, 1988).

However, the contents of the NDCDB are envisioned as neither seamless or scaleless. The archival portion of the NDCDB will be partitioned by series, quadrangles, and categories (Guptill, 1986). These partitions are defined as:

Series: a partition by data content (imagery, elevation matrices, cartographic data) or scale (1:24,000, 1:100,000, 1:2,000,000); interseries topological consistency is not required.

Quadrangles: partitions along latitude, longitude boundaries; matching across boundaries is required.

Categories: a logical subdivision of a series into classes of related data (transportation, hydrography); intercategory topological consistency is required.

The main unit of data collection and revision will remain the map quadrangle. The data are scale dependent, although some thought has been given to creating 2-3 times reduction products (such as 1:50,000-scale map graphics from 1:24,000-scale source or 1:250,000-scale map graphics from 1:100,000-scale source).

U.S. Bureau of the Census

The U.S. Bureau of the Census has developed the Topologically Integrated Geographic Encoding and Referencing (TIGER) system for its use in automating the geographic support system required for the 1990 Decennial Census. The geographic data contained in TIGER consists of a set topological elements with a set of feature directories and lists. The topological elements are represented by 0-, 1-, and 2- cells. Feature lists (containing items such as landmarks, road names, and county names) reference the appropriate set of topological elements that make up the features (Marx, 1986; Kinnear, 1987). Spatial partitioning for the TIGER data base corresponds to county boundaries. Within a county, the data are seamless with all categories of data "vertically integrated" into one planar graph. In addition, more detailed data covering the urban areas has been spliced into the TIGER data base replacing (along 7.5-minute quadrangle boundaries) the data from the 1:100,000-scale maps (Marx, 1987).

U.S. Defense Mapping Agency

As part of its Mark 90 modernization effort, the U.S. Defense Mapping Agency has developed a new data structure called MINITOP (also referred to as the Advanced Mapping, Charting and Geodesy format). The data model underlying the MINITOP structure consists of the following elements: nodes, edges, faces, point feature components, line feature components, area feature components, and features. The nodes, edges, and faces correspond to 0-, 1-, and 2-dimensional topological objects. The point, line, and area feature components are groupings of topological objects of the same dimension. Features are collections of feature components or of other features. Attributes are associated with either the features or feature components. The data base must support a wide variety of products covering various geographic extents across the globe. To support this diversity, the cartographer is presented with a seamless, scaleless window into the data base, although some physical partitioning based on geographic extent and scale is incorporated into the design of the data base archive (C. Kottman, oral commun., 1988).

Institut Géographique National

The Institut Géographique National, France, is in the process of creating two digital cartographic data bases: one with data commensurate with 1:25,000-scale mapping, and one with data commensurate with 1:100,000-scale mapping. The model used in these data bases consists of a set of "elementary objects" corresponding to topological elements and a set of "complex objects" made up of the elementary objects. Descriptive information is associated with the complex objects. The data bases are scale dependent, although, like the Geological Survey case, conceived to support the production of generalized output products with scale reduction factors of two to five. In lieu of an external spatial indexing scheme, the data base is physically partitioned into sets corresponding to a 1-2 km grid (Benard and Piquet-Pellorce, 1986; Salgé and Piquet-Pellorce, 1986; and Salgé, oral commun., 1988).

Landesvermessungamt Nordrhein-Westfalen

The Landesvermessungamt Nordrhein-Westfalen (Surveying and Mapping Agency, North Rhine-Westfalia) is participating in the design of both a digital cadastral map data base (Automatisiertes Liegenschaftskataster - ALK) and a Digital Land Model (Automatisiertes Topographisch-Kartographisches Informationssystem - ATKIS). These activities in the North Rhine-Westfalia region are part of a nationwide project to create the Official Topographic-Cartographic Information System. ATKIS will contain stratified data sets with information appropriate for mapping at scales from 1:5,000 to 1:1,000,000. The ATKIS data model consists of objects that were classified into "point-shaped, line-shaped and area-shaped objects" and are further characterized with attributes. An object has pointers to other objects and to object parts. For each object several attributes of different attribute types and references of different types to other objects are defined (Barwinski and Brüggemann, 1986; Brüggemann, written commun., 1987).

Data Collection Strategies

The results of this brief survey are rather mixed. Those agencies with strong traditions of quad-based standardized mapping (the civilian mapping agencies in the United States, France, and the German Federal Republic) have continued this practice in the construction of their digital data bases. However, the military mapping agencies are supporting a wider range of products over a much greater territory and appear to have adopted a more flexible approach to accomplish their mission. The Census Bureau is in a unique position, acting more like a data user than a producer, modifying and adding value to the base category digital data supplied by the Geological Survey. The Census Bureau has a seamless, scaleless (or at least multiresolution) data base over the areas (counties) of major importance to its mission. The county partitions were the most suitable for Census operations. The conceptual design of TIGER would allow the aggregation of counties to form State or National data bases if desirable, although computer limitations at the time did not allow this aggregation to be implemented (F. Broome, oral commun., 1989).

ENABLING TECHNOLOGY

Seamless, scaleless spatial data bases reflect a nontraditional view of cartography: "cartography as an information transfer process that is centered about a spatial data base which can be considered, in itself, a multifaceted model of geographic reality" (Guptill and Starr, 1984). Spatial data models based on the concept of a set of feature objects that represent aspects of the real world, that is geographic reality, provide a logical framework for seamless, scaleless data bases. But how can these concepts be implemented within a data-base context? Several researchers and commercial firms have implemented spatial data bases within a commercial relational data base. These include the GEOVIEW project (Waugh and Healey, 1987), SYSTEM 9 (Schuch, 1988), and GeoVision (Madill, 1987). However, each of these implementations has had to work around the limited set of data types and operations supported by existing relational DBMS's.

In recent years, computer scientists have sought to extend the capabilities of data base systems, creating a class of "extensible" DBMS's. These systems include university research systems such as EXODUS (Cary and others, 1986) and POSTGRES (Stonebraker and Rowe, 1986), as well as several commercial systems. Several features of extensible DBMS's are of potential use in the implementation of seamless, scaleless spatial data bases. For example, of particular interest are the following design goals of POSTGRES: better support for complex objects; user extensibility for data types, operators, and access methods; and facilities for active data bases (such as alterers and triggers) and inferencing including forward- and backward-chaining.

Using the DLG-E data model and the POSTGRES facilities, speculation on some design characteristics of a seamless, scaleless data base is possible. The seamless requirement to traverse the entire data base and retrieve various elements implies that the DBMS must support abstract data types and user-defined indexes. The scaleless requirement to vary data base resolution implies that the DBMS support multiple representations, user-defined operators, and rules.

The DLG-E spatial objects (points, nodes, chains, and areas) are defined as new atomic abstract data types (ADT) using the POSTGRES abstract data type definition facility (the "define type" command). Creation of these data types allows a further definition of sets of spatial operators (Claire and Guptill, 1982) to work on those data types (using the "define operator" command). Feature objects are each defined as ADT's of type POSTQUEL. This definition allows the feature objects to be represented by a set of shared subobjects, that in DLG-E may either be spatial objects or feature objects. Using field of type POSTQUEL (containing a sequence of commands to retrieve data from other relations) for the DLG-E feature objects will allow for multiple representations of the object (for example, three representations of Dulles Airport). The proper representation from a set of representations (or even a modification of a given representation by a spatial operator) could be invoked using the POSTGRES rule management facilities (Stonebraker and others, 1987). Finally, the "define index" command of POSTGRES allows a user to define secondary indexes on various relations in the data base. The use of external R-Tree or quad-tree indexes, coupled with appropriate access method software, should allow reasonably quick retrievals from large seamless data bases.

CONCLUSIONS

Spatial data users are the driving force toward seamless, scaleless spatial data bases. Data producers would have difficulty justifying such capabilities unless they were needed to satisfy their own internal use (for example, the Census Bureau). GIS users, on the other hand, would probably prefer to view their study area in total, not arbitrarily partitioned by map sheet edges or effected by varying resolution data within the area. With extensible DBMS's, the technology to handle seamless, scaleless data bases is almost at hand. The burden is, therefore, placed on data producers to create future data-base designs that do not preclude users from creating seamless, scaleless versions. Toward this end, the adoption by many mapping agencies of a feature-based, object-oriented data model is a positive step. Assiduous edge-matching across map sheets would be another step forward.

REFERENCES

Barwinski, Klaus, and Brüggemann, Heinz, 1986, Development of Digital Cadastral and Topographic Maps - Requirements, Goals, and Basic Concept: Proceedings of Auto Carto London, Imperial College, South Kensington, London, Vol. 2, pp. 76-85.

Bernard, Antoine, and Piquet-Pellorce, Daniel, 1986, A Workstation for Handling Located Data : PISTIL: Proceedings of Auto Carto London, Imperial College, South Kensington, London, Vol. 1, pp. 166-174.

Cary, M., DeWitt, D., Frank, D., Graefe, G., Richardson, J., Shekita, E., and Muralikrishna, M., 1986, The Architecture of the EXODUS Extensible DBMS: Proceedings of the International Workshop on Object-Oriented Database Systems, Pacific Grove, California, September, 1986.

Claire, R.W., and Guptill, S.C., 1982, Spatial Operators for Selected Data Types: Proceedings of Auto Carto 5, Arlington, Virginia, pp. 189-200.

Fields, Craig, 1978, Beyond "Electronic Paper". Harvard Papers on Geographic Information Systems, Volume Seven: Laboratory for Computer Graphics and Spatial Analysis, Harvard University, Cambridge, Massachusetts, 7 p.

Goodchild, M.F., 1987, Towards an Enumeration and Classification of GIS Functions: Proceedings of International GIS Symposium, Arlington, Virginia, November, 1987, [in press].

Guptill, S.C., 1986, A New Design for the U.S. Geological Survey's National Digital Cartographic Data Base: Proceedings of Auto Carto London, Imperial College, South Kensington, London, Vol. 2, pp. 10-18.

Guptill, S.C., Fegeas, R.G., and Domaratz, M.A., 1988, Designing an Enhanced Digital Line Graph: American Congress on Surveying and Mapping, 1988 ACSM-ASPRS Annual Convention, St. Louis, Missouri, Vol. 2, pp. 252-261.

Guptill, S.C., and Starr, L.E., 1984, The Future of Cartography in the Information Age, in Computer-Assisted Cartography Research and Development Report: International Cartographic Association, p. 2.

Kinnear, C., 1987, The TIGER Structure: Proceedings of Auto Carto 8, Eighth International Symposium on Computer Assisted Cartography, Baltimore, Maryland, pp. 249-257.

Madill, R.J., 1987, Content Management - The Challenge for Geographic Information Systems: Proceedings of International Cartographic Association Conference, Morelia, Mexico, October 12-21, 1987, Vol. 1, pp. 141-147.

Marx, R.W., 1986, The TIGER System: Automating the Geographic Structure of the United States Census: Government Publications Review, Vol. 13, pp. 181-201.

Marx, R.W., 1987, The TIGER System: Six Years to Success: Proceedings of the 13th. International Cartographic Conference, Morelia, Mexico, October 12-21, 1987, Vol. IV, pp. 633-645.

Morrison, J.L., Callahan, G.M., and Olsen, R.W., 1987, Digital Systems Development at the U.S. Geological Survey: Proceeding of International Cartographic Association Conference, Morelia, Mexico, October 12-21, 1987, Proceedings, Vol. 4, pp. 201-214.

Radlinski, W.A., 1974, Untitled Keynote Address in Proceedings of the International Conference on Automation in Cartography, Reston, Virginia, December 9-12, 1974: American Congress on Surveying and Mapping, Falls Church, Virginia, pp. 3-7.

Rhind, D.R., 1988, Personality as a Factor in the Development of a Discipline: The Example of Computer-Assisted Cartography: The American Cartographer, Vol. 15, No. 3, pp. 277-89.

Salgé, Francois, and Piquet-Pellorce, Daniel, 1986, The I.G.N. Small Scale Geographical Data Base (1:100,000 to 1:500,000): Proceedings of Auto Carto London, Imperial College, South Kensington, London, Vol. 1, pp. 433-446.

Schuch, H., 1988, Wild SYSTEM 9: A Perspective for the User: Technical Papers, 1988 ACSM-ASPRS Annual Convention, St. Louis, MO, March 13-18, 1988, Vol. 2, pp. 149-158.

Stonebraker, M.R., and Rowe, L.A., 1986, The Design of POSTGRES: Proceedings of the ACM-SIGMOD International Conference on Management of Data, Washington D.C., May, 1986.

Stonebraker, M.R., Hanson, E., and Hong, C.H., 1987, The Design of the POSTGRES Rules System: Proceedings of IEEE Conference on Data Engineering, Los Angeles, California, February, 1987.

The American Cartographer, 1988, The Proposed Standard for Digital Cartographic Data: Vol. 15, No. 1, 144 p.

Tomlinson, R.F., 1988, The Impact of the Transition From Analogue to Digital Cartographic Representation: The American Cartographer, Vol. 15, No. 3, pp. 249-61.

U.S. Geological Survey, 1986, Digital Line Graphs from 1:24,000-Scale Maps: U.S. Geological Survey Data Users Guide 1, 109 pp.

Waugh, T.C., and Healey, R.G., 1987, The GEOVIEW design. A relational data base approach to geographical data handling: International Journal of Geographical Information Systems, Vol. 1, No. 2, pp. 101-118.

OPTIMAL TILING FOR LARGE CARTOGRAPHIC DATABASES

Michael F. Goodchild
National Center for Geographic Information and Analysis
University of California
Santa Barbara, CA 93106

ABSTRACT

We define a tiling as a partitioning of a spatial database using geographical and thematic criteria. Several options for the ordering of tiles are seen to be analogs of traditional map indexing systems and to fall into a general class of digit interleaving schemes. A general scheme for generating indexes is proposed. Optimality in tile arrangement is defined through an objective function in a form of the quadratic assignment problem. Solutions are described for a number of simple arrangements under assumptions about device behavior and the nature of applications.

INTRODUCTION

The term **tiling** is used in various ways in spatial data handling, but for the purposes of this paper we use it to refer to any system by which a database is partitioned geographically. The existence of such tiles or partitions may or may not be known to the user: in some systems the user manipulates tiles explicitly through a tile manager or map librarian, while other systems hide the management of tiles from the user and present the appearance of a seamless database.

We assume that some form of tiling is inevitable if physical constraints are not to be placed on the potential size of a spatial database. The acquisition of spatial data is limited only by time and cost, and it is often observed that expectations about the coverage and level of spatial resolution of geographical data increase at least as fast as our technical ability to service them. Tiling may be required because of the limited capacity of storage devices, or in order to optimize the efficiency of various algorithms and processes, or to optimize search and retrieval. The distinction between tiling and indexing is clearly somewhat blurred: the term tiling is associated with a physical partitioning of the database, and also implies that manipulation of tiles will occur at the level of the operating system, and that tiles may be distributed over different types of devices.

The traditional analog of tiling is the map sheet, which represents a partitioning of a database into physically separate units. There are parallels therefore between the systems of tile indexing to be discussed below, and the systems devised for indexing map sheets. Traditionally, the map sheets in a series share a common scale and size (ignoring variation due to the earth's curvature and the projection used), which is determined by the constraints of

the printing, distribution and storage technology of paper maps. On the other hand the number of objects represented on the map sheet is not constrained except indirectly by cartographic issues.

In the vector-based domain, the storage requirements of a tile are determined by the number of objects present, and are therefore highly variable even though tiles may be uniform in area. The same is true in the raster-based domain: although the number of pixels may be constant over a set of uniform-area tiles, run-length encoding or hierarchical subdivision will produce a variable volume of data in storage. We define a **fixed** tiling as one in which the area covered by the database is divided into tiles of equal area and shape (usually rectangles) and an **adaptable** tiling as one in which tile size is allowed to vary, probably by hierarchical subdivision, in order to achieve an approximately equal volume of data per tile.

Adaptable tiling clearly has advantages, since the volume of data in each tile can be set to the optimal volume for search, retrieval and other types of processing. Fixed tiling is generally suboptimal, but has advantages in cases where the volume of data changes through time, or is otherwise not known in advance. The costs of restructuring an adaptable tiling in response to new data, in order to maintain optimality, can be substantial.

We assume that in a GIS context it is necessary to store a number of layers or coverages, corresponding to different themes, for the area covered by the database. In the vector domain these coverages will be populated by different classes of objects, and in the raster domain they will consist of layers of pixels. Since a tile can contain any number of classes of objects or layers of pixels, it is clearly possible to reduce the size of a tile either by subdividing its geographical extent, or by subdividing the set of themes.

The purpose of the present paper is to review the concept of tiling from the perspective of optimality. The second section looks at methods of indexing tiles, as a preliminary to the subsequent discussion. The third section proposes a framework for optimization, with specific application to optical stores.

TILE INDEXING

The database of the Canada Geographic Information System (CGIS) (Tomlinson, Calkins and Marble 1976) contains layers of area objects which are partitioned geographically into rectangular tiles of fixed size, known in CGIS as frames. The system of tile indexing was devised by Morton (1966) to ensure that the relative positions of tiles in the database were as far as possible directly related to their relative locations in space. Goodchild and Grandfield (1983) and Mark and Goodchild (1986) proposed measures which could be used to determine the success of different orderings at achieving this objective, and used them to evaluate a number

of standard orders.

The Morton sequence for a square array of n by n tiles can be generated by a simple algorithm as follows. Number the rows of tiles from 0 to $n-1$, and express each row number to base 2. Similarly number the columns from 0 to $n-1$ and express each column number to base 2. The position of each tile in the Morton sequence can be obtained by interleaving the row and column bits, resulting in a binary number between 0 and n^2-1 . More formally, let $\{r_1, r_2, \dots, r_p\}$ denote the ordered set of binary digits forming the base 2 representation of the row number i , $0 \leq i < n$, $2^{p-1} < n \leq 2^p$ and let $\{c_1, c_2, \dots, c_p\}$ similarly represent the ordered set of binary digits forming the column number j , $0 \leq j < n$. Then the ordered set $\{r_1, c_1, r_2, c_2, \dots, r_p, c_p\}$ is the binary representation of k , $0 \leq k < n^2$, the position of the tile (i, j) in the Morton sequence.

If the concept of bit interleaving is generalized to arbitrary bases, it turns out to be identical to many more traditional approaches to indexing tiles or map sheets. Assume as before that the row address of a tile is represented by an ordered set of p digits, but allow each digit s to have a corresponding base x_s . The array is no longer assumed to be square; let m denote the number of rows and n the number of columns. The column address is an ordered set of q digits, digit t having base y_t . We now assume that the bases have been chosen such that the number of rows is equal to the highest number defined by the set of bases:

$$\prod_s x_s = m, \quad \prod_t y_t = n \quad (1)$$

Then the simple row by row sequence starting at row 0 column 0 is generated by setting $p=1$, $q=1$, $x_1=m$, $y_1=n$ and interleaving. Putting the column digit before the row digit will generate a column by column sequence.

A more elaborate example is provided by the GEOLOC geographical referencing system (Whitson and Sety 1987), which indexes every 100 acre parcel in the continental US. The first level of partition consists of 2 rows and 3 columns, each tile being 25 degrees of longitude by 13 degrees of latitude. These tiles are ordered row by row from 1 to 6. At the next level each tile is divided into 26 rows of one half degree latitude and 25 columns of one degree longitude, the area covered by one 1:100,000 USGS quad. Each of these subtiles is given a two-letter designation by concatenating the letter representing the row (a base 26 digit A through Z) with one representing the column (base 25, A through Y).

Each subtile is divided into 4 rows and 8 columns of 7.5' quads, numbered row by row from 1 to 32. At the next level these are divided into 4 rows and 2 columns, designated by assigning the letters A through H row by row. Finally, each of these divisions is divided into 5 rows, lettered A through E, and 10 columns numbered 0 through 9 to produce 50 cells of approximately 100 acres each. An example of a full

designator for a 100-acre parcel (in the Los Angeles area) is 4FG19DC6, or the set of 7 digits {4,F,G,19,D,C,6}, with associated bases {6,26,25,32,8,5,10}. Of these, digits 2, 3, 6 and 7 result from a simple interleaving of row and column digits (digits 2 and 6 from rows, 3 and 7 from columns). Digits 1, 4 and 5 are obtained by first concatenating row and column digits with bases x_s and y_t , respectively and then reexpressing the result with base $x_s y_t$. So in full, the technique requires the interleaving of a set of 5 row address digits with bases {2,26,4,4,5} and a corresponding set of 5 column digits with bases {3,25,8,2,10}, and expressing the result as a set of 7 digits with bases {2×3,26,25,4×8,4×2,5,10}. Generalized digit-interleaving schemes such as these are common in the index systems used for numerous national map sheet series. In general, then, digit interleaving allows tiles to be placed in an order which approximates Morton's earlier objective. The GEOLOC ordering of 7.5' quads clearly comes closer to doing so than a system of ordering alphabetically by state, and alphabetically within state by quad name.

Further generalizations of the digit interleaving concept result when complement operations are allowed. Let the notation r_s^* indicate that the subsequent element in the ordered set is complemented, i.e. its value c_t is replaced by $y_t - c_t$, under certain conditions. For example, c_t might be complemented whenever r_s^* is odd. The example (r_1^*, c_1) now generates a sequence in which each alternate row is reversed (boustrophedon, or the row prime order of Goodchild and Grandfield 1983). A more complex example of complementing operators generates the Hilbert Peano or Pi order.

OPTIMIZATION

Transition probabilities

We now introduce a new notation as the basis for a discussion of optimization. Let i, j denote a tile in the database, consisting of some collection of spatial information, which might be objects or pixels, for some geographical area i and theme subset j . A second such tile/theme combination is denoted by k, l . Now consider the likelihood of requiring both tiles i, j and k, l in some GIS process. For example we might wish to search both tiles for objects having specified attributes, or to display both tiles simultaneously, or to undertake the edgematch operation of matching objects across the common boundary of the two tiles. In a final example we might wish to change the current display from the contents of i, j to k, l .

Let $p(k, l | i, j)$ denote the probability that k, l is the next tile required after i, j , either to replace i, j or to be analyzed simultaneously with it. The transition from i, j to k, l may require change of geographical area, or theme, or both. We assume that change of area is independent of change of theme, in other words that the likelihood of moving from area i to area k is independent of the themes involved, and write $p(k, l | i, j) = p(k | i) p(l | j)$.

The set of themes included in any database is clearly limited, and it would be unreasonable to try to build a spatial database containing all possible themes. Instead the set of themes in a database is limited to those appropriate to the application set. But within a given database, the relationship between geographical subdivision and subdivision of themes is complex. In some systems, each tile includes all themes, requiring relatively small geographical subdivisions. In others, themes are split across several tiles, allowing the geographical area of each tile to be relatively large. CGIS is based on a hierarchical system in which one level of tiling, the 1:250,000 map sheet, contains all themes while a lower level, the frame, contains only one theme. In terms of our notation, if $p(j|j) \gg p(l|j) \forall l \neq j$ then the rational strategy would be to maximize the geographical area of each tile and minimize the number of themes per tile; if the $p(l|j)$ are roughly equal then it is rational to store all themes together and reduce the geographical extent of each tile accordingly. For the set of themes included in the database we suspect that the second case is a more accurate reflection of the needs of most forms of GIS analysis and sets of users, and is the approach used in most of the systems currently available. However a hierarchical approach in which each tile is further subdivided into single themes, and then into geographical partitions of each theme, is commonly adopted in the interests of processing efficiency. If a tile contains all themes we can drop the $p(l|j)$ term and focus on $p(k|i)$, and additionally ignore $p(i|i)$.

It seems reasonable to assume that $p(k|i)$ is a decreasing function of some measure of the distance between k and i . One possibility is therefore to take $p(k|i)$ to be a constant if k and i are neighbors, and zero otherwise. For example, in a rectangular tiling we might take $p(k|i) = 1/4$ for all rook's case neighbors of i . Another is to assume some suitable continuous function of the distance d_{ik} between the centroids of the tiles, such as the negative exponential $\exp(-bd_{ik})$. However in practice we may have access to some additional information which can provide a surrogate for $p(k|i)$. For example in a vehicle navigation system a suitable surrogate would be the probability of the route passing to tile k from tile i , which might be based on the existence of a freeway or on traffic statistics.

Retrieval costs

We assume that all tiles are located on some device, and that there is a cost associated with retrieval of a particular tile which depends on the previous tile retrieved or accessed. Let c_{ik} denote the cost or penalty of retrieving or accessing tile k given that the last tile accessed was i . In the case of a tape, c_{ik} will be dependent on the length of tape separating the tiles, whereas in the case of disk c_{ik} is approximately constant and independent of both i and k . For map sheets stored in cabinets in a map library, we might speculate that c_{ik} shows a complex behavior: low if i and k are adjacent, increasing rapidly with separation if they are almost adjacent, but high and

constant if i and k are separated by more than a few sheets. Certain sheets k are likely easy to find independently of i .

In this paper we are particularly concerned with massive stores with capacity in the gigabyte to terabyte range ($>10^9$ bytes). These include a variety of automatically loading tape systems, in which the individual reel of tape has a capacity on the order of 10^8 bytes. Of particular interest in this paper are massive optical stores (jukeboxes) which provide automatic loading of platters, each with a capacity of 2 gigabytes. Such stores currently provide the only feasible method for efficient storage and retrieval of spatial databases of the size of the USGS's Digital Cartographic Database or the Bureau of the Census's TIGER files.

Such stores are characterized by relatively slow seek times, when the store may be moving its optical read head, sequentially processing tape, or changing tape or platter, and fast bulk transfer rates. For tape stores, c_{ik} is an increasing function of separation on one volume, and roughly constant across volumes: for optical stores, c_{ik} is similarly an increasing function of separation within volume, although numerically much smaller, and high and constant between volumes.

We can now write the expected cost of accessing tile k following tile i as $c_{ik}p(k|i)$, and the expected cost of accessing any tile as its sum over k . The optimum tile arrangement on a given device is that which minimizes:

$$Z = \sum_i \sum_k c_{ik} p(k|i) N_i \quad (2)$$

where N_i is the number of accesses of tile i . The problem of minimizing Z falls into the general class of quadratic assignment problems (Koopmans and Beckmann 1957). In the commonest interpretation c_{ik} is the cost of moving a unit quantity of material between two machines i and k located on a shop floor, $p(k|i)N_i$ is the flow of material between the two machines, and the objective is to locate machines to minimize the total cost of movement.

Although a large literature exists on exact and heuristic solutions to the quadratic assignment problem (Francis and White 1974), we require general solutions which are robust across as wide a set of applications as possible. In the next section we consider the effects of some likely simplifying assumptions.

General solutions

We first assume N_i constant; it would be very difficult to assemble the necessary information on which any other value of N_i might be based, and it is unlikely that the result would be robust across the application set of any given database. We further assume $p(k|i)=1/4$ for all k which are rook's case neighbors of i , otherwise $p(k|i)=0$. Again, such a simple assumption has the advantage that it is likely to be robust across applications.

First let us assume that $c_{ik}=a$ if i and k are adjacent in storage, else $c_{ik}=b$, $b>a$. This leads to a simple solution in which the optimum value of the objective function Z occurs whenever tiles which are neighbors in storage are also neighbors in space. Many arrangements have this property, including the row prime and Hilbert Peano orders, resulting in an objective function value of $(a+b)/2$ per tile, since every tile incurs a cost of a for each of two of its neighbors and b for the other two. On the other hand the expected cost of the Morton order is $(a+3b)/4$ since only one of a tile's four neighbors is adjacent in the Morton sequence. For row order the cost is $[(n-1)a+(n+1)b]/2n$ where n is the number of columns because the tiles at the end of each row have only one adjacent neighbor and thus incur additional cost.

Now assume that tape volumes are used for storage, and that the number of tiles on each volume is such that all of a tile's neighbors are stored on the same volume. It seems reasonable to assume for tape that c_{ik} is a linear function of the separation of the tiles on the tape, that is:

$$c_{ik} = \alpha + \beta |z_i - z_k| \quad (3)$$

where z_i is the location of tile i with respect to the beginning of the tape and α and β are constants. Goodchild and Grandfield (1983) show that for row by row, row prime and Morton orderings of an array of n rows and n columns, the mean absolute difference between a cell's position in the sequence and those of its rook's case neighbors is $(n+1)/2$. In all of these orderings cost is therefore $n^2[\alpha+\beta(n+1)/2]$. Goodchild and Grandfield were unable to obtain a closed-form expression for Hilbert Peano order but their numerical expression gives slightly higher cost. We conclude that Morton order has no advantage over row by row order in this example.

Although we have not been able to prove the general case, we have thus far failed to find a counterexample to disprove the proposition that any ordering of a square array which can be generated by interleaving of digits (including row by row and Morton) has the same mean absolute difference between neighbors of $(n+1)/2$. The proposition is not true if $n \neq m$: for row by row order the result in this case is $[m(n-1)+n^2(m-1)]/(2nm-m-n)$. The question of which order minimizes Z is therefore unresolved in the general rectangular case.

DISCUSSION

The problem of optimal sizing and arrangement of tiles will become increasingly important in the future as spatial databases grow in size. In this paper we have considered one aspect of the arrangement problem, by making certain simplifying assumptions about the objective to be optimized. The questions of optimal size, and of the optimal balance between geographical and thematic subdivision, for different storage devices and applications, remain open.

Clearly it would be easier to define precise objective functions if more information were available on algorithms and patterns of use. On the other hand solutions developed in the absence of such information are necessarily more robust and general. General solutions are likely to be of considerable value in deciding between different storage options for very large databases, as well as in resolving the more specific questions of tile size and arrangement.

REFERENCES

Francis, R.L. and J.A. White, 1974, Facility Layout and Location - An Analytical Approach, Prentice Hall, Englewood Cliffs, NJ.

Goodchild, M.F. and A.W. Grandfield, 1983, Optimizing Raster Storage: An Examination of Four Alternatives: Proceedings, AutoCarto 6, Ottawa, 1:400-7.

Koopmans, T.C. and M. Beckmann, 1957, Assignment Problems and the Location of Economic Activity: Econometrica 25:53-76.

Mark, D.M. and M.F. Goodchild, 1986, On the Ordering of Two-Dimensional Space: Introduction and Relation to Tesseral Principles: in B. Diaz and S. Bell, editors, Spatial Data Processing Using Tesseral Methods, National Environmental Research Council, Reading, UK, 179-92.

Morton, G.M., 1966, A Computer Oriented Geodetic Data Base, and a New Technique in File Sequencing, unpublished manuscript, IBM Canada Ltd.

Tomlinson, R.F., H.W. Calkins and D.F. Marble, 1976, Computer Handling of Geographical Data, UNESCO, Paris.

Whitson, J. and M. Sety, 1987, GEOLOC - Geographic Location System: Fire Management Notes 46:30-32.

The Geographic Database - Logically Continuous and Physically Discrete

Peter Aronson

Environmental Systems Research Institute, Inc.
380 New York Street
Redlands, CA 92373

ABSTRACT

In conventional database terminology a distinction is made between the physical description of the database which refers to how the data is actually organized on the machine, and the logical description of the database which refers to how the data appears to be organized to the user or applications programmer (Martin, 1977). This distinction may be usefully applied to the geographic database as well.

The total sum of data manipulated by a GIS, both locational and descriptive, can be collectively referred to as the geographic database- the map database or library. These geographic databases can vary greatly in quantity of data employed - from the limited project based on a single map sheet and one or two layers of data, to the detailed national database based on thousands of map sheets with hundreds of layers. And while the former case can be handled simply by a single simple data set, the latter case presents additional problems.

For a large geographic database, it is important that the logical view of the data should be continuous - without artificial breaks or storage artifacts. But for the same large geographic databases it is equally necessary that the physical storage scheme used allow for fast random reading and writing of map elements. With current storage technology, such access requires storage of the map data as subsets discrete by locational and descriptive criteria. This paper will discuss the issues raised by such a geographic database architecture and the solutions arrived at in the ARC/INFO GIS.

INTRODUCTION

The tool used by the ARC/INFO GIS to access, manage, and maintain large geographic databases is the LIBRARIAN subsystem (smaller geographic databases are handled by the core GIS in a more ad hoc fashion). The design of the LIBRARIAN subsystem was begun six years ago and made public five years ago (Aronson and Morehouse, 1983). Since that time, the product has evolved considerably. This paper is in part a report on that evolution and in part a discussion of the problems of geographic database creation, maintenance, and access.

The paper is divided into four basic sections: the first section defines the basic problems of the GIS geographic database; the second section details the solutions to those problems used in the LIBRARIAN subsystem; the third section describes the evolution of the LIBRARIAN software over the last five years; and the fourth section considers future directions.

THE GEOGRAPHIC DATABASE PROBLEMS

Geographic databases, like all databases, must be constructed in, alas, a less than perfect world. Data arrives from multiple sources in multiple formats in multiple scales, projections, and with varying data extents; all of which somehow must be integrated into a single geographic database. The geographic database must be organized in such a fashion to make the organization's normal operations sufficiently quick that the organization can use the geographic database to meet its goals. The geographic database must be maintained and updated as required - a task somewhat more complicated than the maintenance of the traditional database.

Very few organizations collect all of the data that they use themselves. Much of the data is either purchased from a commercial data source or, obtained from some government agency. In the Dane County, Wisconsin example described by Chrisman (Chrisman and Niemann, 1985), the seven layers in the database were provided by five organizations: two federal, one state, and two county. This is typical of land records information in this country. In the commercial sector the situation can be even more complicated, since data is often purchased from multiple service bureaus.

This outside data may be supplied in any variety of format, scale, and projection. The areal extent of one supplier (say seven and one-half minute quadrangles) may not match that of another (SMSAs). It may be updated at different intervals (say every two years against every ten). Data integration is a continuing task all through the existence of a geographic database.

Some geographic information systems organize the data themselves; most require that the user make certain decisions on how the system will organize their data. In either case, the data must be organized in a fashion that: allows the data to be useful (that is, supports the organization's access to the geographic database); allows the data to be accessed with reasonable speed; and allows the data to be maintained without great difficulty. There are always trade-offs involved in these decisions - speed versus ease-of-update, ease-of-use versus flexibility, and so forth (anyone who tells you otherwise is a salesman).

Maintenance and update are where geographic databases can be the most difficult. In normal database work, a unit of work is referred to as a transaction. A typical transaction might consist of updating all salary fields of managers in an employee table. This is simple atomic operation in a conventional database management system. In a GIS, geographic database transactions are not so simple.

The type of transaction that we have in GIS geographic database maintenance is what is referred to as a long transaction. In a long transaction, slices of the database (the geographic database in our case) are extracted for update, worked on, then reinserted (Beller, 1988). Where a conventional database transaction lasts (typically) seconds, a GIS long transaction routinely requires days to complete. Simply making the involved data unavailable during the transaction may be unacceptable due to the length of the operation. A more complete solution is required.

Once you've solved the problems involved in building a geographic database, there is still the question of how it is to be used. In general, there

are three purposes for which a geographic database is used: 1) generation of routine products; 2) query; and 3) the generation of ad hoc products.

Routine products, such as yearly forest harvest maps or zoning update maps, are those that are produced according to regular procedures at either known times (spring) or in response to a predictable stimulus (zoning changes). These products are the *raison d'être* for most geographic databases, and the bread and butter of the organizations maintaining them.

The term "query" describes a whole host of applications, including (but not limited to): informal database examination; ad hoc generation of graphics and reports; simple modeling; and examination of geographic database status for management purposes. Their common characteristic is that their subject cannot be predicted in advance, and their output is ephemeral. These are common applications for users such as planners and database administrators. For most organizations, query applications are an insufficient reason to build a geographic database on their own merit, but are considered a valuable secondary benefit from constructing the geographic database.

Ad hoc products lie in a region between query and routine products - like queries, their subject cannot be predicted in advance, but like routine products, there is concrete output. For those organizations with complex responsibilities, these can be an important type of application, even to the degree of justifying the geographic database's existence. For organizations with simpler responsibilities, ad hoc products may be much the same as queries - not vital, but a valuable side effect of having built the geographic database.

THE ARC/INFO SOLUTION

The form that medium-to-large geographic databases take in ARC/INFO is the map library. The tool used to manage and maintain these map libraries is the LIBRARIAN subsystem. Access to map libraries is either via the cartographic and editing subsystems (ARC/PLOT and ARCEDIT) or via copies created by LIBRARIAN.

The basic design principle of the LIBRARIAN is normal use - that is, the LIBRARIAN is designed to support best the operations that are performed the most. The highest priority is hence given to tools that aid in the production of routine products, the next priority to tools for query, and lowest to tools for ad hoc products. Not to say all three kinds of operations are not supported; they are, but map library is designed to give the fastest results in production of the routine product, then query, and then all other operations (the production of ad hoc products is not necessarily slow, just not as optimized).

The map library is a device which allows geographic data to be organized into a large, complex geographic database. Coverages (a digital form of a single topic map) are organized simultaneously in two dimensions -- by subject or content into super-coverages called layers and by location into tiles (see Figure 1).

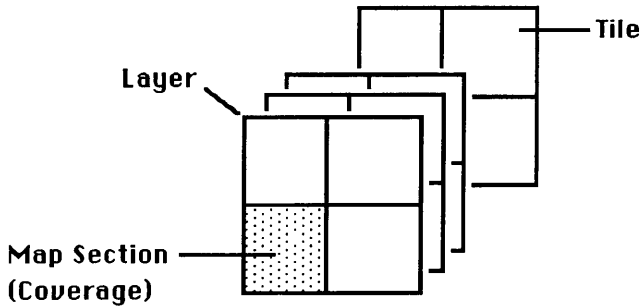


Figure 1: Structure of a Map Library

Tiles. The geographic area represented by a map library is divided into a set of non-overlapping tiles. Although tiles are generally rectangular (e.g., 30' squares), they may be any shape (e.g., counties or forest administration units). Tiles are a digital analogue for the map sheets of a conventional map series. All geographic information in the map library is partitioned by this tile framework. The tile layout is determined by the map library administrator at creation time.

Layers. A layer is a coverage type within the library. All data in the same layer have the same coverage features and feature attributes. Examples of layers are land sections, roads, soil types, and wells. A layer is logically a coverage, but can be physically multiple coverages - the user deals with the layer as single entity although the software may actually deal with multiple entities.

Map Sections. Once a layer has been subdivided into tiles, it consists of a set of individual units called map sections. A map section is simply a coverage as defined previously. Map sections are a physical storage entity, not a logical one.

The tiles are defined in a special INDEX coverage, where each polygon in the INDEX coverage represents a single tile in the library. Layers are defined by defining the feature classes present and the thematic data associated with each feature class.

Map libraries are used for four basic geographic database operations: 1) maintenance (including creation); 2) supplying data for routine products; 3) query; and 4) supplying data for ad hoc products. Each of these topics is discussed below.

Mapbase Maintenance

The basic organizational concept of geographic database maintenance in LIBRARIAN is the long transaction. The software supports long transactions by means of a data object called a named transaction. When a named transaction is created, it owns a set of map sections. These map sections may not be modified except by that particular named transaction. Data is checked out by the named transaction when extracted from the map library, and checked back in when returned after update.

A typical use of a named transaction would be to update a group of street maps. First, the library manager would use the SETTILES and SETLAYERS commands to restrict the tiles and layers affected, like this:

Librarian: LIBRARY URBAN

You have entered transactional library URBAN with MANAGER access.

Librarian: SETTILES LOWERTOWN UPPERFALLS WESTEND

Librarian: SETLAYERS NAME STREET

Having selected the area and topic of interest, the library manager would then begin a named transaction, extract the data (so that it can be edited), then leave LIBRARIAN to do the actual editing, like this:

Librarian: TRANSACTION BEGIN UPROADS/89 Updating STREETS ~

Librarian: in Traffic Zone #02 for 89 road repairs

Librarian: EXTRACT DISSOLVE WORK>DATA>EDIT

Librarian: QUIT

Beginning a transaction marks all the map sections specified by the set layers and set tiles as belonging to that transaction. Performing the extract marks those map sections within the transaction that contain data as checked out. An entry for one of the map sections in the map sections involved in this transaction would read:

TILE:	LOWERTOWN
LAYER:	STREETS
TRANSACTION:	UPROADS/89
STATUS:	OUT

With there being one such entry per map section marked by the transaction.

After leaving LIBRARIAN, the extracted data would be edited. When editing is complete, then the library manager would run LIBRARIAN again, set the proper transaction, re-insert the data, and then end the transaction (assuming it is all done), like this:

Librarian: LIBRARY URBAN

You have entered transactional library URBAN with MANAGER access.

Librarian: SETTILES LOWERTOWN UPPERFALLS WESTEND

Librarian: TRANSACTION SET UPROADS/89

Librarian: INSERT STREETS STREETS

Librarian: TRANSACTION END All done and checked off

Librarian: QUIT

If there had still been data checked out, then the transaction end operation would have failed, with the error message listing the errant map sections.

Routine Products

Routine geographic database products can be divided into two basic categories: those requiring modeling, and those that only require data selection and symbolization. Those that require modeling, LIBRARIAN handles by extracting them from the map library, so that the modeling can be performed in the user's local workspace. This keeps the temporary data sets generated by the modeling process out of the permanent geographic database. Those products that do not require modeling are produced directly out of the geographic database.

As stated above, the basic design principle of LIBRARIAN is normal use. This is most critical in the case of routine products, which are by definition normal use. It is here where the user-defined tiles can be particularly valuable. By selecting a tile grid or tessellation that matches the boundaries required by a majority of the routine products, operations on the geographic database can be extremely efficient. (For a more complete discussion of how the user determines the tile boundaries, see Keegan and Aronson, 1983.)

To support this approach, LIBRARIAN supplies two sets of commands: coverage based commands that use the outer boundary supplied from a specified polygon coverage; and tile based commands that operate on tiles specified by name.

<u>Operation</u>	<u>Coverage based</u>	<u>Tile based</u>
Determine working set	SETCOVER	SETTILES
Extract from database	EXTRACT	GETTILE
Insert into database	INSERT	PUTTILE

The tile based commands operate on tile name and require very little computation or processing. The coverage based commands require calculation of tiles overlapped and the assembly of separate pieces or the splitting into pieces of the data. Hence, this approach yields fast response on the normal use case, but still supports other operations.

For those routine products that don't require modeling, ARC/INFO's graphic output engine, ARCPLOT, is capable of operating directly from the map library. Since ARCPLOT not only displays and symbolizes data, but can produce subsets based on spatial and thematic criteria, change projections, locate by address, and perform simple statistical modeling, a reasonable percentage of products can be produced directly from the map library. And as an added benefit, ARCPLOT serves the front end for cartographic publication product generation - output from ARCPLOT can be sent to PostScript typesetters or to a Scitex graphic production system.

Query

Because the two functions have considerable overlap, the primary query engine for ARC/INFO is its graphic output engine, ARCPLOT. ARCPLOT can produce graphics of all sorts, reports, tables, statistical analysis, and limited derived data sets. Combined with front-end programs written in ARC Macro Language (AML), it can be customized to perform specialized or routine queries. Features can be symbolized and data accessed from tables stored in

ORACLE, INGRES, or SQL/DS, as well as from INFO. For the vast majority of query processes, no actual extraction from the map library is required.

An additional query-like operation is performed in ARC/INFO's coverage editor, ARCEDIT, where data from the map library can be used as a backdrop during the digitizing and editing process. This can help ensure that data involved in a long transaction matches the data still in the map library where desired.

It should be noted that while data is involved in a long transaction, it is still available for query purposes. Given the length of these transactions, it would be undesirable to shut the library down until they were complete.

Ad Hoc Products

Like routine products, ad hoc geographic database products can be divided into those requiring modeling and those that can be produced solely by use of the graphic output engine. Those requiring modeling supply a more interesting problem, as the other case is essentially identical to query.

To support the production of ad hoc modeling products, LIBRARIAN supplies several data extraction options. The area to be extracted can be defined by either the outer boundary of a coverage or by a list of tiles. Data can be extracted by whole tile or clipped to fit a boundary. Features split by the storage scheme can be aggregated to their original form or left divided.

An example of the use of the map library to support the creation of an ad hoc product might consist of the extraction, clipping and aggregation of two layers over an arbitrary region.

Librarian: LIBRARY URBAN

You have entered transactional library URBAN with MANAGER access.

Librarian: SETCOVER RIVER BASIN

Librarian: SETLAYERS PARCELS SOIL

Librarian: EXTRACT DISSOLVE * CLIP

Librarian: QUIT

This operation would create two coverages (a soils coverage and a parcels coverage) that contained all the polygons contained within the river basin, clipped to the boundary of the river basins, with the tile breaks removed. The data extracted from the map library is logically identical to that stored within it, even if it may be stored in a physically different manner.

THE EVOLUTION OF LIBRARIAN

The evolution of the LIBRARIAN subsystem is primarily of interest on account of the lessons learned during the process. LIBRARIAN was formally released as part of version 3.0 of ARC/INFO in early 1985, and has undergone considerable changes in the four years since. Most of these changes have come about in response to user requirements. Nothing matures software like actually being used for real applications.

The LIBRARIAN software described five years ago was a smaller, simpler

system. There were no named transactions, no tile based operations, no access from ARCPLOT or ARCEDIT. It was just the bare bones of a simple map sheet management system. Since that time, every release of ARC/INFO has contained improvements to the subsystem. Described below are some of the changes and the requirements that brought them about.

Tile Based Operations

The original requirement for extraction from the map library was that the user had to have a coverage of the spatial extent of the area to be extracted. However, this is a strange requirement for users who have done a careful tile layout to support their normal use - they usually want one tile at a time, and they know its name. Some of our natural resource users went so far as to produce programs to generate a coverage for extracting whose boundary was identical to a particular tile's border! The same problems applied on insertion as well. Adding these commands made operations far simpler for many of our users.

Graphics and Query Access

The original version of LIBRARIAN included a module called QUERY which was a simple graphics and listing generator that operated on map libraries. QUERY, however, had very limited symbolization capability, and was yet another program to learn. By integrating map libraries into ARCPLOT, the full capability of ARC for graphic production is available straight from the map library. This limits the amount of data that needs to be actually extracted from the map library.

Macro Language Support

The addition of a macro language to ARC/INFO came at version 4.0, released in 1986. At this time the LIBRARIAN subsystem was reorganized in part to make use of AML to write menu-driven front ends more practical. Since then, users have requested (and received) additional functions in AML to support the writing of macros that drive LIBRARIAN, allowing non-technical users to access the map library without really knowing that it exists.

Spatial and Attribute Indexes

While not added specifically to LIBRARIAN, these additions to ARC/INFO very positively impact the use of map libraries. The spatial indexes consist of an improved quad-tree structure which speeds up spatial queries and subsetting. The attribute index is a B+ tree structure that improves thematic selection and scans of lookup tables. These indexes made casual query usable interactively even on the largest map libraries.

Transactions

Transactions were added to deal with highly dynamic libraries, such as a parcel database for a very large city, where there will be many changes per day, and hence some form of collision management is required. In such an organization, there will often be several individuals or even groups responsible for geographic database update, and a mechanism was definitely required to make sure that the process flowed smoothly.

FUTURE DIRECTIONS

While LIBRARIAN is a mature product it is not a senile one. By not senile, I mean that it has not become so bastardized and rococo that further improvements are difficult or even impossible. There are future extensions of LIBRARIAN planned, due to both the ARC/INFO design team's plotting and requests from our users (who as a group are not noted for either shyness or an unwillingness to speak their minds).

Currently a layer and a coverage are much alike. There is no particular reason (aside from some work required) why they couldn't become functionally identical. Anywhere a coverage is accepted, a map library layer would be accepted as well.

One concept we would like to add is that of geographic database views. In SQL, a view is a table derived from another table or tables by way of an expression that may contain joins, selection, aggregation, etc.. A geographic database join would be a coverage or layer that is derived from other coverages or layers via selections, projections and transformations, and possibly spatial joins. Depending on the amount of processing allowed in the view actualization, nearly all modeling could be performed using this mechanism.

CONCLUSIONS

Large geographic databases present problems somewhat different from other types of databases. Maintenance problems may be dealt with by use of long transactions. Production of routine products can be added by structuring the geographic database in such a way as to facilitate efficient operations by the spatial divisions required by those products. Query and cartographic production is facilitated by integrating the geographic database into the graphic output engine. Ad hoc products require flexibility in accessing the geographic database.

The ARC/INFO map library is a data structure designed to meet the above goals. It has been in general use for four years now, and has been upgraded to meet user requirements in that time. It is a mature software product, but not yet a senile one.

REFERENCES

Aronson, P. and Morehouse, S., 1983, The ARC/INFO MAP LIBRARY: A Design for a Digital Geographic Database, Proc. Auto Carto 6, 1983.

Beller, A., Concurrency and Recovery for GIS Databases, unpublished paper, 1988.

Chrisman, N. and Niemann, B., Alternative Routes to a Multipurpose Cadastre: Merging Institutional and Technical Reasoning, Proc. Auto Carto 7, 1985. p. 84-93.

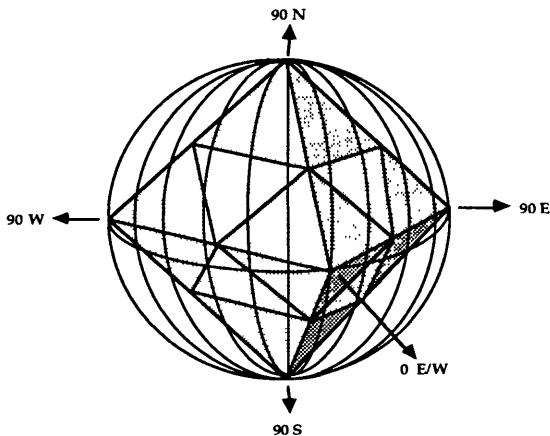
Keegan, H. and Aronson, P., Considerations in the Design and Maintenance of a Digital Geographic Library, Proc. Auto Carto 7, 1985. p. 313-321.

Martin, J. 1977, Computer Data-Base Organization, 2nd edition, Prentice-Hall, New Jersey.

Planetary Modelling via Hierarchical Tessellation¹

Geoffrey Dutton
Prime Computer, Inc.
Prime Park, MS 15-70
Natick, MA 01760
(508) 655-8000

GHD%OASD.Prime.Com@relay.cs.net



Abstract

In encoding, simplifying, amalgamating and intersecting cartographic features, it is useful to know the precision of locational data being processed. It is suggested that representing spatial phenomena in a hierarchical tessellated framework can facilitate documenting the certainty of coordinates and in dealing with its consequences. Objects thus encoded can identify the precision with which they were measured, and can be retrieved at lower degrees of precision, as appropriate. This scale sensitivity is an inherent aspect of quadtree and pyramid data structures, and one which the literature on GIS data quality has yet to address in detail. A specific hierarchical tessellation of the sphere into triangular facets is proposed as a basis for indexing planetary data; Although composed of triangular facets, the tessellation is a quadtree hierarchy. Its geometry is such that its facets are planar, subdivide a sphere naturally and are efficient to address. Methods for generating and manipulating hierarchical planetary geocodes are described.

1 Locational Data Quality

Whatever else they may convey, all spatial data possess coordinate locations. Each geographic entity recorded in a GIS must have an identifiable spatial signature among its properties. As a GIS must be relied upon to integrate and analyze independent collections of spatial data, it should possess means for coping with variability in the quality of coordinate and other data in the features, layers and themes it records, according to their nature, source and purpose. This is usually not possible, hence rarely done.

¹Based on paper originally prepared for Specialist Meeting 1 for the first research initiative of the National Center for Geographic Information and Analysis, Santa Barbara, CA, Dec. 12-16, 1988.

Without the ability to generate spatial inferences, a GIS is little more than an inventory of digitized geographic facts. In order to draw quantitative conclusions about objects in space and time, one must know or be able to estimate the reliability and certainty of the tools and information employed. All too few GIS tools in common use attempt to utilize the scant quality data that their databases may provide. Much has been written about building data quality information into GIS, but few actual systems deliberately do so, and none seem to take its implications seriously. While this state of affairs is not new, it is even more a cause for alarm today than it was five years ago: "We experience difficulty in articulating the quality of information represented in a database principally because we don't understand how to analyze data based on information about its qualities" (Dutton, 1984b).

Mensuration as Modelling. Most GIS enforce a distinction between recording locations and modelling features. Locations are denoted by coordinates, which in turn are associated with features (objects in the real world modelled via some abstraction mechanism). The coordinates pin the features to the Earth at one or more locations, but do not specify how they are encoded. Should coordinates change (due to resurvey, editing or recalculation, for example), this normally has no impact on the features associated with them beyond causing changes in size and shape. Yet, when coordinates change, something important may have happened. We have been so paraDIMEed into fanatically enforcing a dichotomy between the topology and coordinates of cell complexes that we have come to assume that topology alone supplies structure, and there is no structure worth knowing about in a feature's coordinates. This ignores much of the "deep structure" (Moellering, 1982) that geographic data — even coordinates — may be viewed as having. We believe that the coordinates of features indeed have a "depth" component, that this can be modelled via hierarchical tessellation, and that this approach can better characterize uncertainty about cartographic features.

2 Hierarchical Tessellations

Hierarchical tessellations are recursive subdivisions of space-filling cells on a model surface, or manifold. The most familiar group of hierarchical tessellations is the family of data structures known as *quadrees*, square lattices of 2-cells that double their resolution as their number multiplies by four, down to some limit of resolution [see Samet (1984) for a detailed review of the quadtree literature; quadtrees are discussed in relation to GIS by Samet (1986) with a rejoinder by Waugh (1986)]. Other geometries and branching schemes more suitable for modelling global distributions have been proposed or developed (Dutton, 1984a; van Roessel, 1988; Mason and Townshend, 1988; Tobler and Chen, 1986), but few have gained acceptance in the GIS realm. In reviewing and comparing data models for global GIS applications, Peuquet (1988) states:

... a regular, hierarchical spherical tessellation would have many advantages as a global data model. First of all, such a model would retain all of the desirable properties of a planar tessellation including implicit spatial relationships; geographic location is implied by location in the database. Multiple scales and a regular structure are also amenable to rapid search.

Quadtrees were developed to facilitate image processing operations, and for the most part have continued to be oriented toward raster technology. As Waugh (1986) notes, this can be a drawback for GIS applications, which tend to use vector data. Furthermore, while map sheets can be regarded as images and handled as rasters, it is a mistake to think of a GIS as a catalog of maps; while a GIS may manage map data, it can go much further than maps in representing properties of spatial phenomena. As the Earth is neither flat nor a cube, any scheme that is based on subdividing rectangular map images of a planet will fail to provide consistent global coverage (consider how the UTM grid system contorts itself to cover the globe). Cubic quadtrees have been developed to store global data (Tobler and Chen, 1986; Mark and Lauzon, 1986). These have tended to stress storage and retrieval of map and image data (such as segmentation and other conversion tasks), rather than modelling of planetary phenomena. Quadtrees represent a technology in search of applications; planetary modelling is a set of applications in need of technologies. GIS offers an environment where they may connect, provided some basic outstanding issues

are addressed. In a brief but well-informed overview of global database issues, Goodchild (1988) describes the need for research on planetary spatial analysis:

... there is as yet no (spherical) extension of the Douglas-Peucker line generalization algorithm, and only limited literature on the generation of Thiessen polygons and polygon skeletons. There is no spherical version of point pattern analysis, and no literature on spatially autocorrelated processes. It is clear that much research needs to be done in developing a complete set of spatial analytic techniques for the spherical case.

We feel that the paradigm of *geodesic tessellations* may provide keys to unlock some of these problems, by enabling higher-order data modelling capabilities that vector, raster, quadtree and hybrid data structures can draw upon to handle planetary data in a consistent fashion, as the remainder of this paper will attempt to demonstrate.

Polyhedral Tessellations. Rather than developing data structures (either raster or vector) to encode a map — or even a map series — one can base one's efforts on the requirement to describe an entire planet, then subdivide the model into tiles of useful size. This will at least assure that (unlike UTM sheets) tiles will fit together regularly and consistently. The most obvious choices for a basis for tessellation are the five platonic solids; other regular polyhedra (such as a cubeoctahedron, rhombic dodecahedron or rhombic tricontahedron) can be used (and have been for map projections), although not all are capable of self-similar, recursive tessellation (the shape of facets may change when subdivided). Given a basis shape that can be indefinitely subdivided, it is necessary to select one of several alternative tessellation strategies. Triangular facets, for example, may be subdivided into 2, 3, 4, 6 or 9 triangular tiles. In some of these tessellations the shapes of tiles may vary, in others their sizes may vary, or both size and shape may vary. This is the same problem that designers of geodesic domes face; they tend toward solutions in which struts and connectors are as uniform as possible, as this expedites the manufacture and assembly of geodesic structures. The great majority of geodesic domes break down each facet into either four ("Alternate") or nine ("Triacon") tiles (Popko, 1968).

3 A Geodesic Planetary Model

We have been investigating a method of modelling planets based on triangular tessellation of an octahedron, in which each facet divides into four similar ones; this yields successive levels of detail having 8, 32, 128, 512, 2048, ... facets overall, or 1, 4, 16, 64, 256, ... facets per basis octant. The cover page illustrates the basic form and orientation of the model, and figure 1 its development. Table 1 itemizes statistics for this hierarchy and its linear and areal dimensions if Earth-sized. In Table 1, column 1 indicates the hierarchical level of breakdown, column 2 ($=4^{\text{level}}$), and column 3 ($=2^{\text{level}}$), respectively indicate the number of triangular facets and edges that partition an octant at each level. Columns 4 and 5 itemize the linear resolution and unit area each level has on a sphere 4,000 km in radius; approximate distances and areas are given for the spherical wedges defined by the polyhedral facets. Column 6 specifies the number of bits needed to identify facets, reflecting the "cost" of precision.

We shall not attempt to justify this particular tessellation as an optimal one; the scheme does appear, however, to strike a balance between geometric utility, scale sensitivity and computational cost as a way model the surfaces of spheroids. As its vertices are at right angles, an octahedron readily aligns itself to cardinal points in a geographic world grid; subsequently-introduced vertices are easily computed, as they bifurcate existing edges (as shown on the cover page). The breakdown generates eight quadtrees of facets; the structure may be handled as if it were a set of rectangular region quadtrees. However, as their elements are triangular rather than square, many of the geometric algorithms devised for rectangular quadtrees will not work on such datasets without modification.

We call this spatial data model a *Quaternary Triangular Mesh (QTM)*. The remainder of this section will explore some of QTM's geometric, informational and computational properties. The sections to follow will focus on the use of QTM in modelling spatial

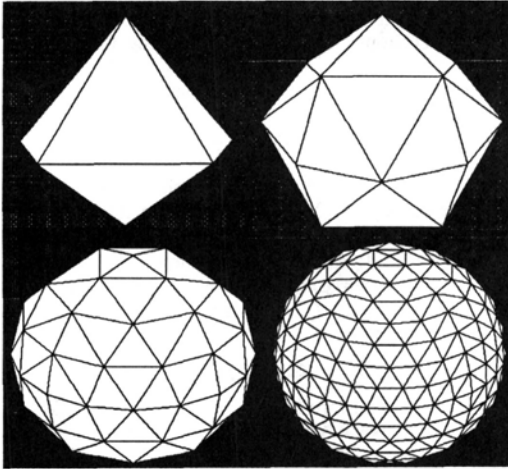


Figure 1 (left):
Development of Quaternary
Triangular Mesh to level 3
on a basis octahedron

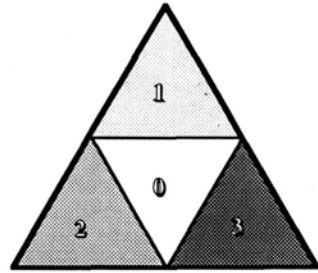


Figure 2: QTM Facet Numbering

Table 1: Planetary Octahedral Triangular Quadtree Statistics
for 1 to 24 Hierarchical Levels (per octant)

LEVEL	FACETS	DIVISIONS	RESOLUTION	FACET AREA	CODE BITS
1	4	2	1444 Km	15,924,500 KmSq	2
2	16	4	722 Km	3,981,125 KmSq	4
3	64	8	361 Km	995,281 KmSq	6
4	256	16	180 Km	248,820 KmSq	8
5	1,024	32	90 Km	62,205 KmSq	10
6	4,096	64	45 Km	15,551 KmSq	12
7	16,384	128	23 Km	3,888 KmSq	14
8	65,536	256	11 Km	972 KmSq	16
9	262,144	512	6 Km	243 KmSq	18
10	1,048,576	1,024	3 Km	61 KmSq	20
11	4,194,304	2,048	2 Km	15 KmSq	22
12	16,777,216	4,096	705 M	3,796,696 MSq	24
13	67,108,864	8,192	352 M	949,174 MSq	26
14	268,435,456	16,384	176 M	237,294 MSq	28
15	1,073,741,824	32,768	88 M	59,323 MSq	30
16	4,294,967,296	65,536	44 M	14,831 MSq	32
17	17,179,869,184	131,072	22 M	3,708 MSq	34
18	68,719,476,736	262,144	11 M	927 MSq	36
19	274,877,906,944	524,288	6 M	232 MSq	38
20	1,099,511,627,776	1,048,576	3 M	58 MSq	40
21	4,398,046,511,104	2,097,152	1 M	14 MSq	42
22	17,592,186,044,416	4,194,304	69 Cm	4 MSq	44
23	70,368,744,177,664	8,388,608	34 Cm	9,052 CmSq	46
24	281,474,976,710,656	16,777,216	17 Cm	2,263 CmSq	48

entities, and how this might address problems of precision, accuracy, error and uncertainty in spatial databases. Throughout, the discussion's context will remain fixed on exploring *QTM* as a geodesic, hierarchical framework for managing and manipulating planetary data. The work reported here stems from a scheme (appropriately known as DEPTH) for storing digital elevation data using polynomial coefficients organized as quadtrees (Dutton, 1983); this was subsequently recast into a global hierarchical triangular tessellation for terrain modelling called GEM (Dutton, 1984a). The tessellation geometry employed for *QTM* is similar to that proposed by Gomez Sotomayor (1978) for quadtree representation of digital terrain models adaptively split into triangular facets. We

use a different numbering scheme, and embed our model in a spherical manifold rather than a planar one (although for many *QTM* computations, a projection is best employed).

QTM as geocoding. In a *QTM* tessellation, any location on a planet has a hierarchical address, or *geocode*, which it shares with all other locations lying within the same facet. As depth in the tree increases, facets grow smaller, geocodes grow longer and tend to become more unique, being shared by fewer entities. A *QTM* address identifies a particular triangular facet at a specific level of detail; that triangle's vertices are fixed on the *QTM* grid, covering a definite patch on the planet. Each such facet can be subdivided (by connecting its edge midpoints) into four similar ones, numbered 0 through 3, as illustrated by figure 2; we refer to the four children of each facet as its *tiles*. Each tile thus generated can be identified by a 2-bit binary number, so that $2L$ bits (or $L/4$ bytes) are needed to specify a *QTM* address at L levels of detail. *QTM* addresses therefore consist of variable-length strings of 2-bit numbers, for example *0311021223013032*. Such identifiers lend themselves to being represented by base 16 numbers, having $L/2$ hexadecimal digits; the 16-level *QTM* address *0311021223013032* is, in hex notation, the (32-bit) number *3526BICE*. To relate this to a more familiar context, *QTM* Addresses at level 16 provide the same order of resolution as LANDSAT pixels. Refer to column 6 of Table 1 for the size of binary identifiers at various *QTM* levels of resolution (divide by four to obtain the size in hex digits).

QTM as geometry. To identify exactly where on earth *QTM* hex geocode *3526BICE* (or any other) lies, one must know the specific method for assigning numbers to *QTM* facets that was employed to construct the geocode. While there are a number of ways to do this, few of them seem useful. The *QTM* tessellation always generates a triangle for each vertex of a facet plus one triangle at its center; we always number corner triangles 1, 2 or 3, and designate the central triangle as zero. This scheme has a convenient property: any number of zeros may be appended to a *QTM* address without affecting its geographic position. While trailing zeros do not modify the location of a measurement, they do signify its precision. We shall return to discuss this property later on.

Having fixed the central triangle as facet 0, we must then assign each of the remaining ones as 1, 2 or 3. Noting that triangles point either upwards or downwards, we identify the orientation of facets as either *upright* or *inverted*: An *upright facet* has a horizontal base with an apex above it, while an *inverted facet* has a horizontal base with an apex below it. All four octants of the northern hemisphere are upright; all four of the southern hemisphere are inverted. Tessellating an octant generates three outer tiles (numbered 1, 2, 3) sharing its orientation, and an inner one (tile 0) having opposite orientation. Let us designate the apex of each triangle (regardless of N/S orientation) as *node 1*, which locates *tile 1*. Nodes 2 and 3 thus define the endpoints of the octant's equatorial base; we can assign them arbitrarily but consistently, thus defining where *tile 2* and *tile 3* are located within each octant, as figures 2, 3 and 7 show. When we do this, we find that the 8 tiles numbered 1 cluster about the north and south poles, and that tiles numbered 2 and 3 lie on the equator. We arbitrarily fix node 2 (hence four of the tiles numbered 2) at the equator (0° N/S) and the Greenwich Meridian (0° E/W), and another (with its four surrounding tiles) at the antipode (180° E/W). Finally, each of the points where the equator intersects longitudes 90° E and 90° W collocate four octant vertices (and tiles) numbered 3, fully defining the numbering of nodes and facets for the first *QTM* level. Figure 3 diagrams this ordering for a sphere and for an octahedron.

QTM as addressing. A depth-first ordering of *QTM* geocodes traces a specific pattern in the process of enumerating an octant's facets. This pattern represents a memory map, delineating the sequence in which geocodes are ordered in computer storage. The compactness of this arrangement helps one map point coordinates to memory addresses which are close to those of nearby points. Exploiting this property can simplify the problem of spatial search from a 2-dimensional procedure to a 1-dimensional one. The pattern generated by visiting successive *QTM* addresses is the set of self-similar, self-intersecting curves shown in figure 4. *QTM* location encoding is clearly a form of spatial indexing; not only are geocodes systematically ordered into quadrees, they have the property that numerically similar *QTM* geocodes tend to lie in close spatial proximity to

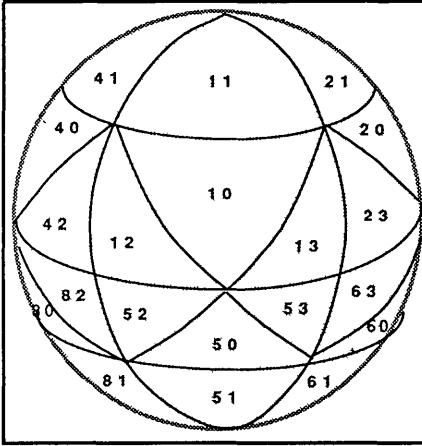


Figure 3a: First-order QTM tessellation of a sphere, showing facet numbering

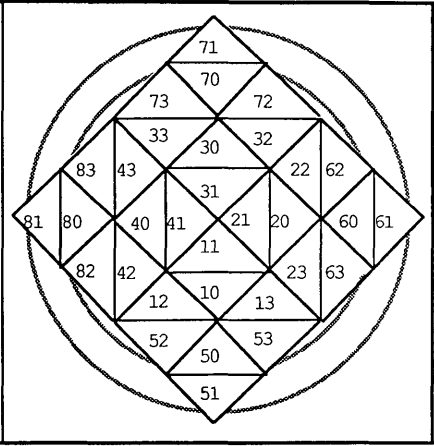


Figure 3b: First-order QTM tessellation of an octahedron, unfolded from S pole

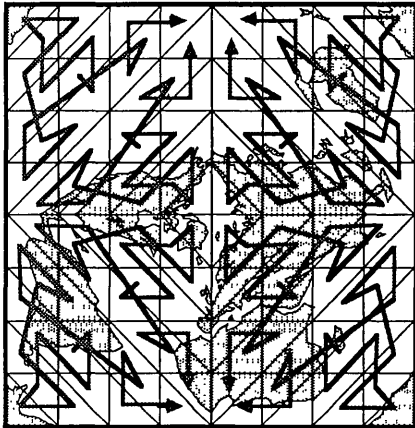


Figure 4: Second-order QTM Code Sequencing (memory map order). ZOT projection.

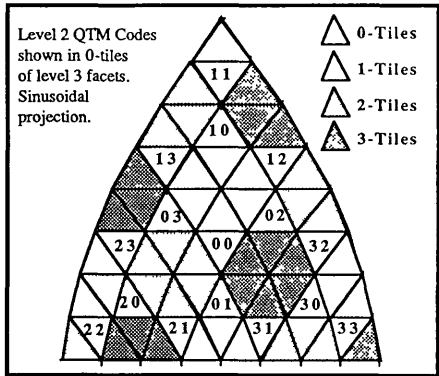


Figure 5: Pattern of least significant digits of QTM codes, forming hexagonal clusters (attractors)

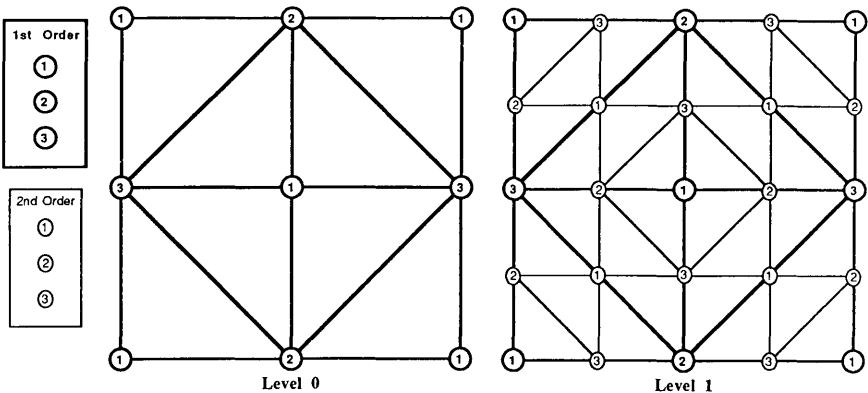


Figure 6: Octa and first level QTM Attractor (node) numerology; Child nodes are numbered 6 - (a+b), where a and b are the nodes of the parent edge.

one another. Furthermore, as a consequence of the numbering pattern described in section 3 above, facets at the same level having *QTM* codes terminated by the digits 1, 2, and 3 form hexagonal groups of six triangles regularly arrayed across the planet; those ending in 0 are isolated triangles filling gaps in the hexagonal pattern. Figure 5 is an equal-area mapping of this pattern for third-order facets for northern hemisphere octants 1 and 3.

This numbering pattern has properties worth noting. The centerpoint of each hexagonal cluster of tiles is a vertex in the *QTM* grid shared by each tile in that group. This nodal point may be thought of as a locus of attraction, or *attractor*, to which nearby observations gravitate. Only tiles numbered 1, 2 or 3 are attracted to such nodes; 0-tiles serve as their own attractors. Once an attractor manifests itself, its basis number will persist in place at all higher frequencies. Space in the vicinity of an attractor is affected as by gravity; the larger an attractor (the shorter its path to its root node), the stronger is its influence. Higher order attractors have smaller ranges of influence than lower order ones, and consequently exhibit less locational uncertainty. Three interlocking triangular grids of hexagons result from this; they cover 75 percent of the planet, with 0-tiles occupying the remaining triangular patches. Figure 6 illustrates the development of attractors; when an edge is bifurcated, a new node appears; we number it as $6 - (a + b)$, where a and b are the basis numbers of the parent nodes.

Aliasing, Attraction and Averaging. As an alternative to mapping locations to *QTM* facets, one may consider *QTM* grid nodes as their loci. By aliasing tiles to nodes, a higher degree of spatial generalization results. It differs from allocating coordinates to facets in that it averages as well as partitions observations into sets. Node aliasing provides a key to dealing with a particularly vexing consequence of many region quadtree schemes, the unrelatedness of adjacent high-order tiles that share an edge also separating lower-order facets. Each level of a quadtree isolates facets (and any values that may be recorded for them) into four subtrees. Whether values are built up from area estimates or obtained via progressive point sampling, discontinuities can occur between sub-branches of the tree simply due to the placement and orientation of the sampling grid. While this can be mitigated by smoothing the resultant grid of values (as demonstrated for terrain in Dutton, 1983), this solution is inelegant and should not be necessary.

We can better understand node averaging by conducting the following thought experiment: Sample a continuous surface, such as topographic relief, assigning *QTM* addresses to a set of 3D point observations, aliasing all source locations which happen to fall into the same *QTM* facet to the same elevation, as there is only one value stored per facet.¹ Let us assign the elevations of the 0-tiles to their centroids, and assign averages of the elevations of proximal 1-, 2- and 3-tiles to their common *QTM* nodes, as figures 5 and 6 show.² We thus obtain a mesh of triangles, the vertices of which have fully-defined latitudes, longitudes and elevations. The edges of the mesh connect *QTM* nodes to the centroids of their facets. A surface defined by these facets will, in general, be smoother (exhibit less aliasing) than one defined by interconnecting the centers of adjacent atomic tiles. Furthermore, because node elevations are spatially symmetric averages, a surface thus defined is relatively stable under translation and rotation (unlike an unaveraged *QTM* coverage, or any quadtree for that matter); its contours would not appreciably change were the orientation of the *QTM* grid to be incrementally shifted.³

4 Computational Considerations

Tessellation methods have long been advocated as ways to partition and index spatial data. The majority of this work seems to be oriented toward decomposing vector and

¹ This can be done by stratifying elevations and recording the changes between strata as attributes of facets, as described in (Dutton, 1983) and (Dutton, 1984a). As it is difficult to avoid aliasing elevations, the surface as encoded may be excessively quantized.

² This requires algorithms which, given a *QTM* facet ID, can identify the *QTM* node to which it aliases, and the IDs of the other facets that converge at that point.

³ While we are confident of this, a formal proof (that node averaging results in a more representative and stable sampling of spatial attributes) remains to be constructed.

raster databases into tiles of fixed or varying size and content (Weber, 1978; Vanzella and Caby, 1988) for access in a GIS. Such approaches lead to various hybrid data structures, in which an overview is provided by a tessellated component and details furnished by the vector and raster primitives. Conceptually, this seems little different than storing data as electronic map sheets of equal or differing sizes. We feel that geodesic tessellations have considerably greater modelling power than has been exploited to date.

Known and unknown properties. *QTM* addresses could replace coordinates in a georeferenced database. When their length is allowed to vary, the accuracy of the positions they encode can be conveyed by their precision. Therefore, the number of digits in a *QTM* geocode may be used as a parameter in processing the coordinates and/or attributes it represents. This permits the precision of coordinate points to be independently specified, and in turn allows analytic procedures to make more informed judgements in modelling the behavior of spatial entities. Describing features at varying precision may or may not result in greater efficiency: As presented here, the *QTM* model does *not* specify how spatial entities are defined, how storage for them is structured or how to manipulate *QTM* elements. While we understand how to perform certain operations on *QTM* geocodes, we know little about how to optimize data structures or processing based on *QTM*'s tendency to cluster nearby locations in memory, or how to best take advantage of the facet-node duality that we have called attractors.

Evaluating spatial data at *QTM* grid nodes might simplify spatial analysis tasks. For example, the need to identify and remove slivers following spatial overlay might be lessened by filtering the coordinates of the features of input coverages via *QTM* tessellation. As all coordinates in the neighborhood of a node are mapped to its location, slight variations in otherwise identical vector strings will tend either to vanish or to alias into structured caricatures of themselves. A related property of *QTM* that begs for application is the behavior of geocodes as identical digits are appended to them: the *QTM* codes *031*, *0311* and *03111*, for example, all alias to the same attractor, hence can represent the same point. Appending more *ones* does not define a new node, it simply constricts the locus of influence for the attractor defined by *031*, adding precision to it (as do trailing zeros; see sect. 3). However, should some other digit follow such a group (e.g., *031112*), a new attractor will come into play, changing the locus of the geocode. It turns out that for Octant 1, the area dominated by the attractor of *QTM* geocode *031* (also an attractor of five other 3-digit geocodes) is in the USSR, centered in the Caucasus between the Black and Caspian Seas. Respecifying the *QTM* code *031111* as *031112* results in shifting to another attractor 50 km away.

QTM in context. There is an increasing amount of literature and interest concerning the properties and computational geometry of hierarchical tessellations. The subject appears to connect many branches of knowledge and goes back many years, involving disciplines as diverse as crystallography, structural engineering, design science, computer science, solid geometry, lattice theory, fractal mathematics, dynamical systems and geography. One particularly relevant source of information concerning the properties of hierarchical tessellations is a group of research fellows and fellow travelers based at the British Natural Environment Research Council (NERC) (Mason and Townshend, 1988). Most of this work is less than five years old, and tends to view the subject matter in a general, theoretical fashion.¹ As befits workers in a field that knows no bounds, the NERC group has coined the adjective *tesseral* to characterize hierarchical tessellations; it is rooted in the Greek word *tessera* – the tiles used in making mosaics. *QTM* is a tesseral construction.

One of the more interesting aspects of the tesseral perspective is the possibility of developing special arithmetics for manipulating elements of hierarchical tessellations. This was demonstrated for the generalized balanced ternary (GBT) system, a hexagonal tessellation developed at Martin Marietta in the 1970's as a spatial indexing mechanism

¹ The NERC papers are solidly in the tradition of fugitive spatial analysis literature that is GIS's birthright: The Michigan geographic community's discussion papers; *Harvard Papers in Theoretical Geography*; Dave Douglas' subroutine library; ODYSSEY (a fugitive GIS); the Moellering Commission's reports, and multitudes of other government research studies, reports and documents.

(Lucas, 1979). GBT's numbering system allows direct computation of properties such as distances and angles between locations without manipulating coordinates (van Roessel, 1988). Other tessellations have related arithmetics, some of which have been explored in the Tesseral Workshops (Diaz and Bell, 1986). Such an arithmetic could be developed for the *QTM* tessellation should none already exist.

Polyhedral operations. One common objection to polyhedral data models for GIS is that spherical geometry is quite cumbersome (in the absence of tessellar arithmetic operators), and that for many applications the spherical coordinates that describe polyhedra require frequent conversion to and from cartesian coordinates. Because planar geometrics are generally much more straightforward than spherical ones, it is almost always easier to compute quantities such as distances, azimuths and point-in-polygon relations on the plane than on the sphere. The former may involve square roots and occasional trig functions, but rarely to the degree involved in geographic coordinates, where spherical trigonometry must be used unless rather small distances are involved or approximations will suffice. Polyhedral geometry, being faceted, is locally planar but globally spherical. What can be considered "local" varies, according to the projection employed (for plane coordinates) or the type and level of breakdown (for tessellations).

Perhaps the most basic polyhedral operation is the derivation of facet addresses (geocodes) from geographic coordinates (or its inverse). This involves recursive identification of triangular cells occupied by a geographic point, appending each identifier to the location code already derived. This process has been named *trilocation*, and is described in Dutton (1984a). One of the simplest trilocation algorithms derived to date for triangular tiles determines a tile's ID by comparing the squared distance from the test point to the centroid of its facet's 0-tile and each of the three outer ones until the closest one is found (this usually takes 2 or 3 squared distance comparisons per level). If performed using geographic coordinates, great circle distances are needed, but if done in the planar domain cartesian distances will suffice (in neither case need square roots be extracted, as we are interested in ordering distances, not in their absolute magnitudes).

5 Conclusions

Effective spatial analysis in a GIS environment seems to require detailed information about data quality, not just statistical error summaries. It is a truism that numerical representations of map data – particularly coordinates – can convey the illusion of accuracy simply because numbers tend to be represented at uniform, relatively high precision. No GIS in general use parameterizes the precision of coordinate data to reflect its inherent accuracy or precision. As a result, intelligence potentially useful for spatial analytic and cartographic decisions tends not to be utilized, complicating procedures and engendering uncertain, *ad hoc* analyses. Solving this problem is critical and calls for the development of new models of spatial phenomena, as Chrisman (1983) explains:

Space, time and attributes all interact. Quality information forms an additional dimension or glue to tie those components together. Innovative data structures and algorithms are needed to extend our current tools. No geographic information system will be able to handle the demands of long-term routine maintenance without procedures to handle quality information which are currently unavailable.

A recurring problem, and one that we create for ourselves, involves the very idea of coordinates; it is generally assumed that coordinates exist in nature, when in fact they are rather artificial notations for spatial phenomena. Features in a GIS don't actually *have* coordinates, coordinates are in fact *ascribed* to them as are other attributes. Too much of the work in spatial error handling has been devoted to tools that deal with coordinates rather than with spatial entities; too little consideration has been given to exploring alternative spatial paradigms. A polyhedral, tessellar perspective might provide this, by offering a unified framework for representation, an inherent sensitivity to scale and new mechanisms for dealing with spatial error and uncertainty. Geodesic modelling offers the GIS community a rare opportunity to create more effective tools for addressing some of the multitude of problems, both local and global, now facing us and our planet.

References

- Bell, S.M., B.M. Diaz, F. Holroyd and M.J. Jackson, 1983: Spatially referenced methods of processing vector and raster data, *Image and Vision Computing* 1, no. 4, 211-20.
- Chrisman, N.R., 1983: The role of quality information in the long-term functioning of a Geographic Information System, *Proc. Auto-Carto Six*. Ottawa: Canadian National Committee for the 6th Int. Symp. on Automated Cartography, pp 303-312.
- Diaz, B.M. and Bell, S.B.M., 1986: *Proc. of the Tesselar Workshops*, 13-14 Aug 1984 and 22-23 1986. Swindon, Wilts UK: Natural Environment Research Council.
- Dutton, G., 1983: Efficient Encoding of Gridded Surfaces, *Spatial Algorithms for Processing Land Data with a Microcomputer*. Cambridge, MA: Lincoln Institute for Land Policy Monograph.
- Dutton, G., 1984a: Geodesic Modelling of Planetary Relief, *Cartographica* 21, nos. 2 & 3. Toronto: U. of Toronto Press, pp 188-207.
- Dutton, G., 1984b: Truth and its Consequences in Digital Cartography, *Proc. 44th Annual Mtg. of ASP-ACSM*, 11-16 March. Falls Church, VA: ACSM, pp 273-283.
- Gomez Sotomayor, D.L., 1978: Tessellation of Triangles of Variable Precision as an Economical Representation of DTM's, *Proc. Digital Terrain Models Symposium*, May 9-11, St. Louis, MO. Falls Church, VA: ASP, pp 506-515.
- Goodchild, M., 1988: The Issue of Accuracy in Spatial Databases, *Building Databases for Global Science* (H. Mounsey and R. Tomlinson, eds.). London: Taylor & Francis, pp 31-48.
- Lucas, D., 1979: A multiplication in N-space, *Proc. Amer. Math. Soc.* 74, no. 1, pp 1-8.
- Mark, D.M. and J-P Lauzon, 1986: Approaches to quadtree-based geographic information systems at continental and global scales, *Proc. Auto-Carto 7*. Falls Church, VA: ASPRS/ACSM, pp 355-364.
- Mason, D.C. and J.R.G. Townshend, 1988: Research related to geographical information systems at the Natural Environment Research Council's Unit for Thematic Information Systems, *Int. J. of Geographical Info. Systems* 2, no. 2 (April-June), pp. 121-142.
- Moellering, H., 1982: *The Challenge of Developing a Set of National Digital Cartographic Data Standards for the United States*. Nat. Comm. for Digital Cartographic Data Standards, Rpt. no. 1, pp 1-15.
- Peuquet, D., 1988: Issues Involved in Selecting Appropriate Data Models for Global Databases, *Building Databases for Global Science* (H. Mounsey and R. Tomlinson, eds.). London: Taylor & Francis, pp. 66-78.
- Popko, E.F., 1968: *Geodesics*. Detroit: University of Detroit Press.
- Samet, H., 1984: The Quadtree and Related Hierarchical Data Structures, *ACM Computing Surveys* 16, no. 2 (June), pp. 187-260.
- Samet, H., 1986: Recent Developments in Quadtree-Based Geographic Information Systems, *Proc. 2nd International Symposium on Spatial Data Handling*, Seattle, WA, 5-10 July. Williamsville, NY: International Geographical Union, pp. 15-32.
- Tobler, W. and Zi-tan Chen, 1986: A Quadtree for Global Information Storage, *Geographical Analysis* 18, pp 360-71.
- van Roessel, J.W., 1988: Conversion of Cartesian Coordinates from and to Generalized Balanced Ternary Addresses, *Photogrammetric Engineering and Remote Sensing* 54, no. 11 (November), pp 1565-1570.
- Vanzella, L, and S. Cabay, 1988: Hybrid data structures, *Proc. GIS/LIS '88* vol 1. Falls Church, VA: ASPRS/ACSM, pp 360-372.
- Waugh, T.C., 1986: A response to recent papers and articles on the use of quadtrees for geographic information systems, *Proc. 2nd International Symposium on Spatial Data Handling*, Seattle, WA, 5-10 July. Williamsville, NY: International Geographical Union, pp. 33-37.
- Weber, W., 1978: Three types of map data structures, their Ands and Nots, and a possible Or, *Harvard Papers on GIS* 4. Reading, MA: Addison-Wesley.

Use of the 1:2,000,000 Digital Line Graph Data in Emergency Response

Hoyt Walker
Lawrence Livermore National Laboratory
University of California
Livermore, California 94550

ABSTRACT

Environmental emergencies often have effects that are distributed over the earth's surface. As a result, maps are usually the most effective way to portray the impact of an emergency. The Atmospheric Release Advisory Capability (ARAC) at Lawrence Livermore National Laboratory is an emergency response organization that utilizes computer-assisted cartography. ARAC provides real-time assessments of the consequences of atmospheric releases of radioactive material. The products of this service are isopleths of the material concentration plotted over a base map of geographic features.

Because ARAC's commitments encompass the entire United States, the ability to produce base maps anywhere in the United States is very important. At present ARAC is using data derived from the United States Geological Survey's 1:2,000,000 Digital Line Graph (DLG) database to meet its small-scale mapping needs. The DLG data set contains much of the information needed to serve in this emergency response application. However, certain enhancements are required to produce the necessary base maps. To create a data set suitable for ARAC, several preprocessing steps are needed. These include transforming the coordinate system, extracting relevant features as individual entities, correcting coding errors, and matching the edges along adjacent sectional files.

INTRODUCTION

Maps play an important role in many fields of endeavor. They are indispensable tools for identifying and representing locations, distributions and spatial variations. Any entity that is near or on the surface of the earth can be mapped at its position with respect to other objects. Collections of entities and extended phenomena are often mapped as distributions. The portrayal of spatially-varying phenomena on a map facilitates the comprehension of their variation. Thus maps can be applied to any field of inquiry involving location or spatial variation. Cartography, the art and science of making maps, is most closely allied with geography, which includes the task of studying spatial variation and its underlying causes. However, numerous other fields place heavy reliance on maps.

The recent application of computers to cartography has profoundly influenced the field. Performing mundane tasks in map-making, such as the calculation of map projection coordinates, is one example of the computer's utility. While the influence of the computer is not limited to the reproduction of tedious manual operations at a higher speed, the rapid creation of new maps tailored to evolving requirements permits the use of cartography in applications where traditional methods would be far

too slow. Operational emergency response is an area where automated cartography plays such a role.

An example of an emergency response system where heavy reliance is placed on maps produced by computer is provided by the Atmospheric Release Advisory Capability (ARAC) located at Lawrence Livermore National Laboratory. ARAC uses numerical models to estimate the dispersion of radioactive material in the atmosphere. The execution of these models requires the integration and manipulation of various kinds of spatial data. Measured and derived data must be presented coherently so as to provide a clear picture of an evolving problem during an emergency response. To provide a locational reference for these presentations, base maps, composed of the transportation network and hydrography along with various political and administrative boundaries, are required. Much of the data used in these base maps is generated by a tablet digitizing system. However, such digitization is time-consuming and expensive to complete. As a result, attempts are being made to take advantage of digital map data produced by government agencies in creating base maps. The 1:2,000,000 Digital Line Graph (DLG), produced by the U.S. Geological Survey (USGS), is an example of such digital map data. While 1:2,000,000 DLG data is too coarse for many applications, it does meet a number of the requirements for ARAC mapping. Consequently, substantial effort has been expended in incorporating this data into the ARAC system. This paper will discuss the ARAC project and its mapping needs, the 1:2,000,000 DLG database, and some of the problems in the data set that, when corrected, make the information more useful.

ARAC

ARAC is an emergency response system capable of addressing accidents in which radioactive material is released into the atmosphere. The ARAC system, which resides at the Lawrence Livermore National Laboratory (LLNL), is composed of computer systems, numerical models, data-gathering systems, data analysis techniques, and highly trained operational personnel (Dickerson and others 1983; Dickerson and others 1985). ARAC has responded to such real-world events as the Three Mile Island (Knox and others 1981) and Chernobyl (Dickerson and Sullivan 1986) reactor accidents, as well as the COSMOS satellite reentries.

ARAC models

ARAC relies on a number of numerical models that simulate the transport and diffusion of material through the atmosphere. Of these many programs and models, the primary model exists as a stream of five codes that are executed in a regular cycle as a problem evolves. These codes are three-dimensional and incorporate the effects of topography and complex meteorology. Meteorological data from around the world is received in real-time from the Air Force Global Weather Central. Elevation and map databases are built as part of the maintenance of the ARAC system. Detailed elevation data is derived from the Defense Mapping Agency's Planar data in the U.S. (Walker 1984) and from their Level I Digital Terrain Elevation Data for areas outside the U.S. Coarse elevation data for long range transport modeling is extracted from the U.S. Geophysical Data Center's ETOPO5 data set. This body of information, along with descriptions of the accident scenario, is integrated by the assessor in order to select various model parameters. This complete data set is used by the models to produce real-time assessments of dose distribution as well as short-term projections of the future distribution. Mapping has an important role to play in *this process* (Walker 1985). First, maps are used by the assessors to validate all of

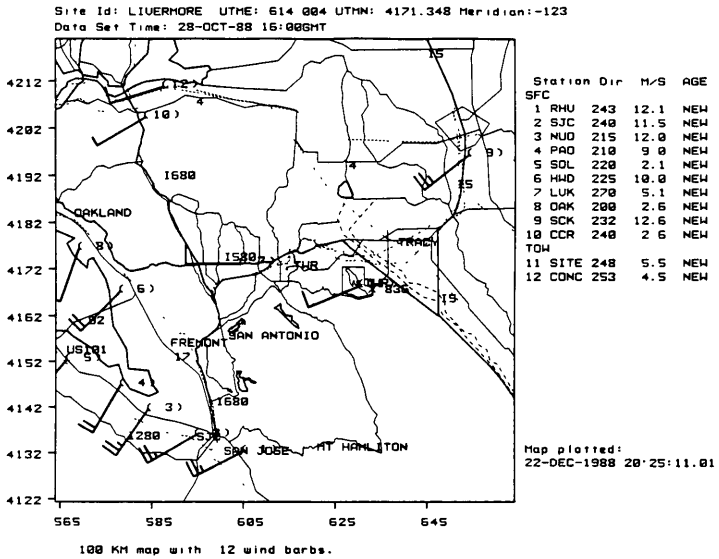


Figure 1. A map of wind barbs showing wind speed and direction at the measurement locations for validating the quality of incoming meteorological data.

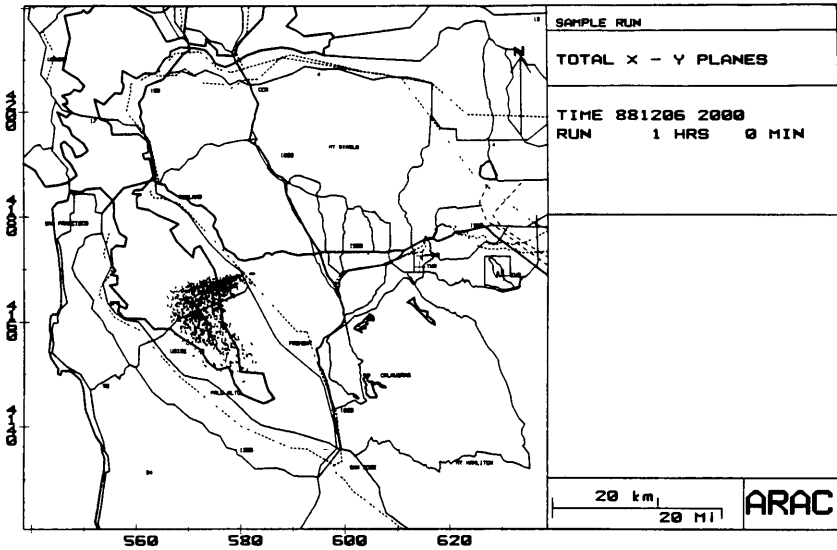


Figure 2. A map produced by a model showing marker particles used to simulate transport and diffusion in the atmosphere.

the input data quickly and accurately (see Figure 1). Second, each model in the stream produces a series of picture frames, many of which are maps, that allow the assessor to follow the evolution of the modeling process (see Figure 2). Lastly, the results of an ARAC assessment are distributed as a graphical image, i.e., a map is drawn that incorporates isopleths of material concentration or dose (see Figure 3).

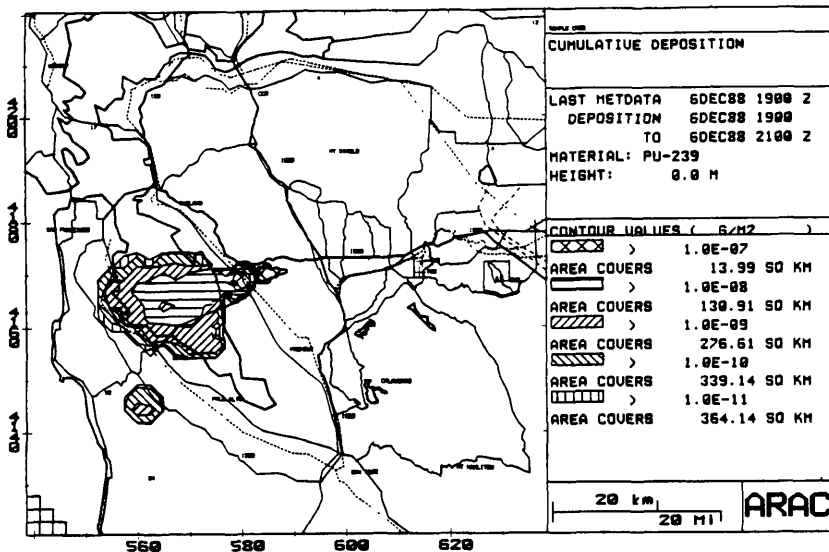


Figure 3. A typical ARAC product showing the deposition pattern of released material which is Plutonium-239 in this example.

Mapping requirements

The operational and emergency response nature of the ARAC system places a number of basic constraints on the mapping component of the system. For example, the time limitations that exist in an emergency along with the need to easily integrate the maps with other computer-based information imply that all mapping must take place on the computer. The time constraints also imply that all or nearly all of the digital map data must already exist in a easily accessible form before an emergency occurs. Most of the map data currently used by ARAC is produced on a primitive tablet digitizing system. This system is based on old hardware and has numerous fundamental limitations. Thus, the current system makes it difficult to make quality digital maps and is also quite slow. The quality problems are readily apparent in the sample maps shown in Figures 1,2 and 3. Some of these difficulties will eventually be alleviated with the design or acquisition of a new digitizing system. However, tablet digitizing is likely to remain too time-consuming to complete after the start of an emergency. As a result, digitizing is only effective for sites that subscribe to ARAC's service and for which preparation is possible before the occurrence of a release.

While ARAC supports a substantial number of Department of Energy and Department of Defense sites, most incidents to which ARAC has responded have been at unexpected locations anywhere on the globe for which there was no specific preparation. The spatial extent over which the effects of a release are a concern range from less than ten kilometers to an entire hemisphere. Consequently, complete coverage of the globe with digital map data at a wide range of scales is desirable. The volume of such data needed to meet these broad requirements is enormous and clearly beyond the digitizing capabilities of a small project such as ARAC. Instead, ARAC relies on the national mapping agencies for the creation of digital map data that can meet its requirements.

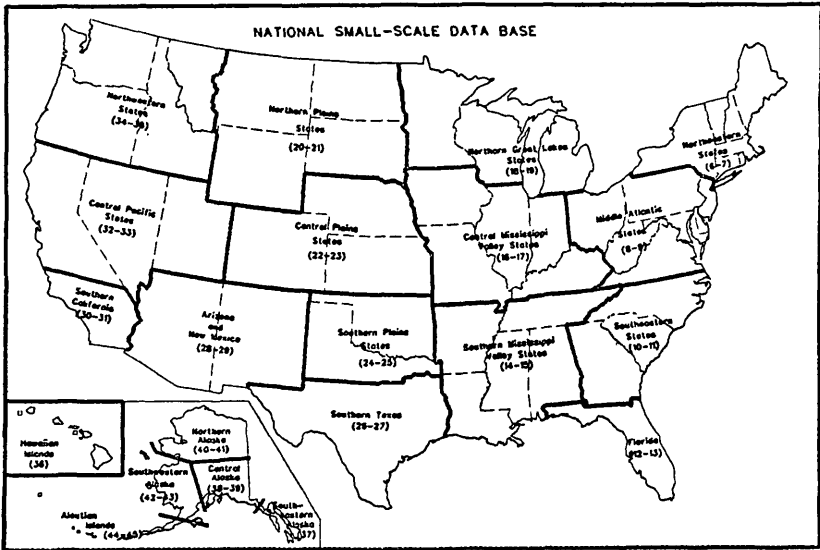


Figure 4. The coverage of the U.S. by reference map sheets in the National Atlas and in the 1:2,000,000 DLG database (from USGS).

At present, ARAC utilizes digital data to produce small-scale maps for long range transport problems. A data base produced by the National Center for Atmospheric Research is used for very small-scale maps of global or hemispheric extent. For larger-scale problems up to about 1:2,000,000 in scale, ARAC has made use of World Data Bank II (Porny 1977) which was produced by the Central Intelligence Agency. While this data set provides global coverage, it has some disadvantages including the lack of transportation networks, limited attribute coding and no topological structure. To improve ARAC's mapping capabilities in this scale range, the USGS 1:2,000,000 DLG data set has been examined closely.

1:2,000,000 DLG

The 1:2,000,000 DLG data was produced in response to a perceived "urgent multi-user requirement for a national small-scale digital cartographic data base" (Stephens and others 1979). To meet this need, the USGS chose to digitize, in vector format, the General Reference Maps from *The National Atlas of the United States of America* which were drawn at the scale of 1:2,000,000. These reference maps, published in 1970, cover the entire U.S. in 21 separate sheets (15 for the conterminous U.S.). The breakdown of the U.S. into these sheets is illustrated in Figure 4. Digitization began in 1979 and the complete database was available in 1984. In some cases, the information that was digitized was updated because the original information was gathered in the late 1960s.

Structure

The digitizing methodology is reflected in the structure of the database. The map separates for each of the different sheets were digitized into separate files. The map separates comprise seven overlays containing the following classes of information: (1) roads, (2) railroads, (3) streams, (4) waterbodies, (5) political boundaries,

(6) administrative boundaries, and (7) cultural features. Cultural features on these maps are civilian and military airports symbolized as points. The choice of vector representation as opposed to raster grids reflects the line orientation of the various overlays which consist of linear features, area features with linear bounds, or point locations. Such information can be stored more efficiently as vectors than in grids.

To support more advanced cartographic and geographic applications of this data, USGS implemented a topological structure for each overlay in a sheet along with two non-topological data structures. Thus, there are three DLG formats referred to as DLG-1, DLG-2 and DLG-3, with DLG-3 supporting topological data. This paper only considers the DLG-3 format. Topological information explicitly defines spatial relationships among cartographic objects such as connectedness and adjacency (Peucker and Chrisman 1975). While such relationships can be derived from the geometry of the data, this is usually difficult and expensive. The topological information normally associated with vector cartographic data is based on graph theory; this is reflected in the name *Digital Line Graph* used by USGS for their distribution format for cartographic data. A good overview of the DLG format is presented in Luman (1987).

Line elements (*chains*, in the terminology adopted as a cartographic standard in 1987 (Morrison 1988)) are the central component of the DLG format. Such a line element is composed of a directed locus of points bounded by a beginning node and an ending node. The line element includes pointers to the areas on its left and right as well as attribute information describing what the line represents. Nodes in the DLG format are only associated with a position. Areas are associated with attribute information and a position (not necessarily within the area). Thus the topological structure is simple as well as efficient in terms of storage because only lines contain pointers to the other topological elements. This is desirable for data transfer; however, such a structure requires more computation due to searches through the set of lines to find the lines connected to a node or adjacent to an area. While topological structuring is provided within each overlay of a map sheet, the current DLG format does not provide for topological connections between the different overlays of a sheet (*vertical integration*) or between adjacent map sheets (*horizontal integration*). In addition to the absence of topology between overlays and map sheets, no effort was made to match the geometry of the line elements between overlays and map sheets. For example, streams typically do not end precisely at shorelines and roads are discontinuous at sheet boundaries.

Applications

Published descriptions of applications of the 1:2,000,000 DLG database appear to be rare. The USGS has used the data to reproduce the original Middle Atlantic States Reference Map at 1:2,000,000 scale from the digital data (Dixon 1985). Edge discrepancies were resolved in producing the finished map. The Federal Emergency Management Agency (FEMA) is using this data as the basis for the mapping component for a dial-up system for emergency response called the Integrated Emergency Management Information System (IEMIS) (Jaske 1985). IEMIS was designed to assist FEMA decision-makers during such emergency situations as floods and hurricanes. It also has atmospheric modeling capabilities suitable for releases affecting small areas. IEMIS uses the 1:2,000,000 DLG without addressing the problems of horizontal or vertical integration.

ARAC USE OF DLG

As mentioned above, ARAC perceives the 1:2,000,000 DLG as playing an important role in its wide-ranging mapping requirements. This data has a number of superior characteristics as compared to WDB II, its main competitor in this scale-range. These characteristics include intra-overlay topological structuring, detailed attributing coding, and a consistent and high-quality map source. While these factors make the DLG data set a clear choice for coverage of the U.S., substantial processing has been necessary to tailor the data to the specific needs of ARAC. This processing has the purpose of integrating the data consistently into the larger system that already exists at ARAC as well as improving and supplementing the data where possible. A number of the more interesting processing steps are discussed below.

Feature identification

One of the more obvious shortcomings of the 1:2,000,000 DLG data is the lack of feature identification. For a number of the overlays, there is no information distinguishing individual features. For example, lakes and islands in the waterbodies overlay as well as the data in the streams overlay are not named. They are classified according to their longest dimension or length. In some ARAC applications, it is helpful to name important features such as lakes. Consequently, interactive graphics software was developed to allow operators to select and name those lakes and islands which could be unambiguously identified on a current road atlas composed of state maps of somewhat larger scale than the DLG data. Smaller lakes were typically left unnamed. The task of naming streams was judged to be too time-consuming to attempt at this time. In the case of the roads overlay, the attribute information is sufficient to identify most of the important roads. However, the low-priority given to feature identification in the development of this database is reflected in a substantial number of attribute coding errors. These errors include both classification and numbering mistakes. Because ARAC can be called on to respond to transportation accidents, it is important to have road identification be as accurate as possible. As with the waterbodies, interactive software was developed to allow examination and correction of all mis-labeled roads. The same road atlas used for naming waterbodies was used to verify road names.

Horizontal integration

Since the areas affected by releases are unlikely to fit nicely within a prescribed map sheet boundary, it is desirable to remove any sheet-to-sheet discontinuities. As a result, the creation of seamless map overlays from the DLG data was identified as an important goal. Interactive software has been developed to allow an operator to identify features that are to be edge-matched. The method of matching is based on the following two ideas. First, the difference in location between two boundary nodes is an approximate measure of the absolute location accuracy of the DLG map sheets. Thus, it is reasonable to shift the lines within the range defined by the endpoints. Second, it is important to maintain the topological relationships between elements in both of the sheets being matched. Therefore, no positional changes are made beyond the internal nodes of the two lines being matched.

In the standard method of matching, the distance along the line from the internal node to the boundary node to be matched is determined for both lines. The ratio of these lengths specifies the location of a point located along the line segment between the two boundary nodes. This new connection point determines how much each of the boundary nodes must be shifted to achieve a match. The shift of the

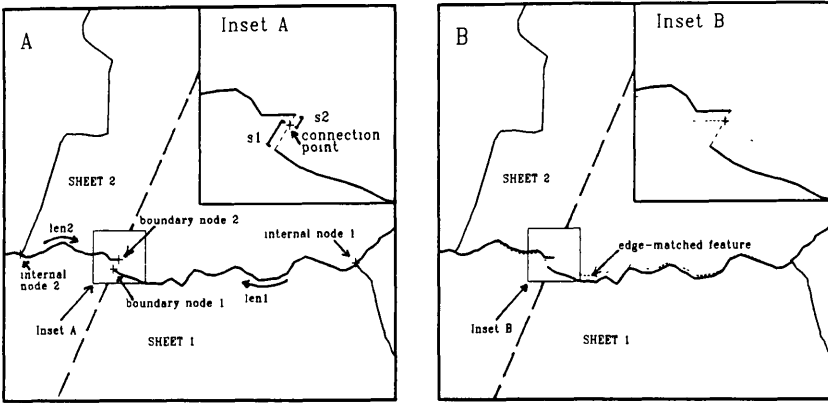


Figure 5. The geometry for matching a linear feature is demonstrated in (A) and (B). The two boundary nodes in (A) mark the points which should match across a sheet boundary. The internal nodes, within the areas of the two sheets, are to be left unchanged by the process. $len1$ and $len2$ are the lengths along the lines from the internal nodes to the boundary nodes in sheets 1 and 2, respectively. The matched lines will meet at the connection point shown in the inset. The connection point is chosen so that $s2/s1 = len2/len1$. The shift required to match the boundary points decreases linearly to zero for the other points along the line as the internal node is approached. The results of this process are shown in (B).

other points between the boundary node and the internal node is the boundary node shift linearly scaled by the relative distance along the line. This linear scaling is chosen so that no shift occurs at the internal node and the full shift occurs at the boundary node (see Figure 5). In other words, the magnitude of the shift at the boundary endpoint decreases linearly to zero as the internal node is approached. This approach normally produces a smooth fit with the shortest line being moved less than the longer line. Other methods are provided to allow some flexibility in handling unusual geometries such as allowing adjustment to only one line instead of both lines.

Vertical integration

There are a number of situations where the absence of vertical integration is problematic. For example, the fact that the streams overlay and the waterbodies overlay are not integrated implies that stream extensions (continuations of streams passing through lakes which allow drainage continuity if the lakes are not plotted) do not necessarily begin and end at lakeshores. In a few extreme cases involving small lakes, the stream extension does not lie within the lake at any point. Related problems occur where streams intersect shorelines and where major rivers (double line streams) intersect the coastline. It was decided to correct the geometry of these problems as much as possible without any operator intervention. The goal was to match intersection points without changing the location of any points on the line if the problem could be corrected by merely shifting the classification of a few points.

Using the stream extension into a lake as an example, software was written to automatically find the intersection of the continuous stream feature with the lakeshore. From this it is possible to determine which points are in the lake and belong to the stream extension and which are outside the lake forming part of the actual stream.

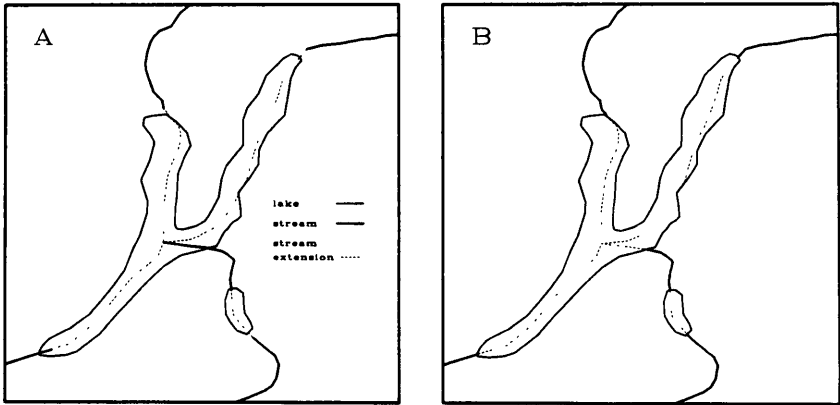


Figure 6. The mislocation of the change from a stream to a stream extension is shown in (A). The corrected alignment appears in (B).

The appropriate points are placed into the appropriate lines and the node is moved to the intersection point. In this case, no locations are changed except for the node (see Figure 6). Situations where neither the stream or the stream extension intersect the lake are not addressed. In the case of a stream intersecting a coastline or a lakeshore where no extension continues into the lake, two situations exist. One, the stream terminates within the waterbody, in which case the stream is truncated at the intersection point. No existing points are moved, only a few are eliminated. If the stream falls short of the shoreline, then the last line segment of the stream is extended until the shoreline is intersected unless a point on the coastline is closer, in which case that point is added to the end of the stream. Thus, vertical integration is accomplished with a minimal change in the preexisting geometry.

Other processing which reflects more specific requirements of the ARAC system includes translation to binary format, coordinate transformations and splitting the seamless map into quadrangles of standard size to match other existing ARAC databases. A regional map produced from the ARAC version of the DLG data is shown in Figure 7.

CONCLUSIONS

The USGS 1:2,000,000 database is a useful source of map information for some applications. Its utility can be improved to the extent that the problems of feature identification along with horizontal and vertical integration are addressed. USGS is aware of these problems and limitations (Guptill 1986) and is currently working on an enhanced DLG format that will allow these issues to be addressed (Guptill 1988). We encourage the USGS to actively continue their work in these directions and hope that eventually all their digital cartographic data will reflect these improvements.

Future work in ARAC mapping will center on the acquisition and integration of the new 1:100,000 DLG product which has recently been made available by USGS. The scales most often used in ARAC responses range from 1:100,000 to 1:500,000. As a result, data extracted from the 1:100,000 DLG should meet the majority of ARAC's need for base maps. Many of the problems associated with this database are

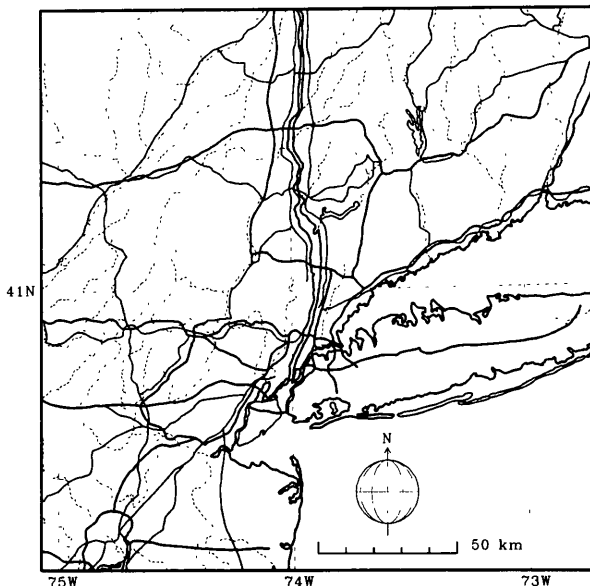


Figure 7. A sample map of the area around New York City showing part of the roads, waterbodies and streams overlays.

related to its size, but these appear to be surmountable with the use of laser disks. Application of line simplification algorithms may be required to produce manageable quantities of data at scales near 1:500,000. Digital cartographic databases form an important component of the ARAC system and improvements to existing databases and the development of new databases are eagerly awaited.

This work was performed by the Lawrence Livermore National Laboratory under the auspices of the U.S. Department of Energy under contract No. W-7405-Eng-48.

REFERENCES

- Dickerson, M.H., Gudiksen, P.H., and Sullivan, T.J., 1983, *The Atmospheric Release Advisory Capability*, Lawrence Livermore National Laboratory Report UCRL 52802.
- Dickerson, M.H., Gudiksen, P.H., Sullivan, T.J., and Greenly, G.D., 1983, *ARAC Status Report: 1985*, Lawrence Livermore National Laboratory Report UCRL 53641.
- Dickerson, M.H. and Sullivan, T.J., 1986, *ARAC Response to the Chernobyl Reactor Accident*, Lawrence Livermore National Laboratory Report UCID 20834.
- Gorny, A.J., 1977, **World Data Bank II, Volume 1 — North America, and General User Guide**, National Technical Information Service, PB-271 869.
- Guptill, S.C., 1986, *A New Design for the U.S. Geological Survey's National Digital Cartographic Data Base*, **AutoCarto London**, 2, pp. 10-18.
- Guptill, S.C., Fegeas, R.G., and Domaratz, M.A., 1988, *Designing an Enhanced Digital Line Graph*, **Technical Papers, ACSM-ASPRS Annual Convention**, pp. 252-261.
- Jaske, R.T., 1985, *FEMA's Computerized Aids for Accident Assessment*, **Proceed-**

ings of an International Symposium on Emergency Planning and Preparedness for Nuclear Facilities, International Atomic Energy Agency, pp. 181-204.

Knox, J.B., Dickerson, M.H., Greenly, G.D., Gudiksen, P.H., and Sullivan, T.J., 1981, *The Atmospheric Release Advisory Capability (ARAC): Its Use During and After the Three Mile Island Accident*, Lawrence Livermore National Laboratory Report UCRL 58194.

Luman, D.E., 1987, *Applying USGS Digital Line Graph Data in a Microcomputer Environment*, *The American Cartographer*, 14, 4, pp. 321-343.

Morrison, J.L., ed. 1988, *The Proposed Standard for Digital Cartographic Data*, *The American Cartographer*, 15.

Peucker, T.K., and Chrisman, N., 1975, *A Cartographic Data Structures*, *The American Cartographer*, 2, 1, pp. 55-69.

Stephens, M.J., Domaratz, M.A., and Schmidt, W.E., 1979, *The Development of a National Small-Scale Digital Cartographic Data Base*, **Proceedings of the International Symposium on Cartography and Computing, (AUTO CARTO IV)**, ACSM, ASP, pp. 345-352.

Walker, H., 1984 *Spatial Data Requirements for Emergency Response*, Lawrence Livermore National Laboratory Report UCRL 91263.

Walker, H., 1985, *Spatial Data Handling Capabilities in Emergency Response*, **Geographic Information Systems in Government**, 1, pp. 295-307.

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

USE OF A GEOGRAPHIC INFORMATION SYSTEM TO EVALUATE THE POTENTIAL FOR DAMAGE FROM SUBSIDENCE OF UNDERGROUND MINES IN ILLINOIS

Carol A. Hindman and Colin G. Treworgy
Illinois State Geological Survey
Natural Resources Building, 615 East Peabody Drive
Champaign, Illinois 61820

ABSTRACT

This paper describes the use of geographic system information (GIS) technology to evaluate the risk of damage to structures from mine subsidence in Illinois. Since the early 1800s, about 3000 underground mines (all but 40 of which are abandoned), have been operated in Illinois to recover coal, minerals, and earth materials. Mine subsidence, the sinking of ground over a collapsed mine, can cause damage to homes and other structures. Maps and tables created with the GIS are used to show the coincidence of underground mines with urban areas and to estimate the number and total value of housing units exposed to subsidence risk. The GIS provides capabilities that solve challenging technical aspects of this project, including compilation of data and the synthesis and presentation of large, diverse data sets. Digitizing, in conjunction with other computer software, provides efficient means for encoding mine locations from source materials having a variety of scales, formats, and degrees of cartographic accuracy. Buffer zones around mine boundaries compensate for uncertainties in the locations of mine boundaries and show surface areas that could be affected by subsidence. Boundaries of urban land are buffered to allow for potential growth. A map library data structure provides efficient handling of large data sets on land cover and mines. The flexibility of the GIS will make it relatively easy to update the study with future census, land cover, and mine data.

INTRODUCTION

This paper describes the use of geographic information system (GIS) technology to help evaluate the risk of damage to structures from mine subsidence. The GIS provides a number of capabilities that help to solve the challenging technical aspects of this project, including the compilation of data and the synthesis and presentation of many large, diverse data sets.

Mine subsidence, the sinking of the ground surface after the collapse of an underground mine, can take place gradually over a large area, or can be quite sudden, opening as a pit at the surface (DuMontelle et al., 1981). This ground movement can result in damage to overlying structures and loss of property value. In Illinois, mine subsidence has occurred over all types of underground mines. Most subsidence events are related to coal mines because of the large number of coal mines and their proximity to urban areas. The largest subsidence event to date, however, was over a lead and zinc mine. More than 2660 underground coal mines have operated in Illinois since 1810; all but 30 are now abandoned. Another 350 underground mines have operated to extract clay, flourspar, lead, zinc, dolomite, limestone, ganister, and tripoli; all but 10 of these mines are abandoned.

Damage caused by "ground movement" is not insured under conventional property insurance. With the inception of the Illinois Mine Subsidence Insurance Fund (IMSIF) in 1979, Illinois became the second state in the country to provide protection against mine subsidence damage to structures. IMSIF reimburses private insurance companies for claims paid for mine subsidence damage.

IMSIF needed information that could be used to evaluate their potential exposure to claims for damage due to mine subsidence. Initially, the only information available was the percentage of each county undermined by coal mines. This information was of little value because in some areas mines are directly under urban development while in other areas mines underlie land having no insurable structures, such as water bodies and cropland. IMSIF also needed to know what areas are undermined by non-coal mines, as there has been mine subsidence over lead and other mines.

This study provides statistics on the proximity of mined areas to urban development and housing. The product of primary interest to IMSIF is a tabulation by township showing acreage of mined areas underlying and adjacent to urban areas, the approximate number of housing units undermined, and the approximate value of those housing units. This information can be derived relatively quickly and cheaply from existing digital data sets: coal mines (Treworgy et al., 1988), land cover (Loelkes et al., 1983; Fegeas et al., 1983), and housing (U.S. Department of Commerce, 1980; Geographic Data Technology, Inc., 1982; Donnelley, 1986). The only data that had to be compiled and digitized were outlines of non-coal mines.

The project was divided into two tasks: 1) to compile and digitize the data on non-coal mines, and 2) to merge the mine information with the data on land cover and housing and present it for IMSIF to analyze.

DEVELOPMENT OF A DIGITAL DATABASE ON NON-COAL MINES

The development of a digital database on non-coal mines involved two problems that were solved by GIS techniques: 1) compilation of mine outlines and mine shaft locations from source maps having a variety of scales and degrees of cartographic accuracy, and 2) documentation of uncertainties of mine location, orientation, and configuration to be tracked and properly considered in later modeling.

Original mine maps were the preferred source for the compilation and digitizing of mine boundaries. We found maps that varied from page-size to wall-size and were drawn on paper (sometimes folded), linen, and tracing paper. Some maps had no scale or reference points. Other maps had incomplete mine boundaries or boundaries that were drawn before mining operations ceased.

Hard copies of original mine maps were made from microfilm of original mine maps acquired through the Federal Office of Surface Mining. Large maps were divided onto two or more microfilm frames. Locations for mines without outlines were taken from maps and legal descriptions in publications and from shaft and mine tunnel symbols on USGS 7.5-minute quadrangles. Although there were 29 different scales ranging from 1:120 to 1:63,360, the majority of maps were at the scales of 1:2400, 1:4800, and 1:24,000.

Compilation of Mine Outlines into a Digital Database

Three basic methods were used to enter the mine locations or outlines into the database: 1) digitizing directly from the mine map, 2) transferring the mine outline to a mylar overlay of a 7.5-minute quadrangle and digitizing the mylar, and 3) using a computer program to convert legal descriptions to X-Y coordinates. Maps in good condition that had at least four reference points (section corners or 1/4-section corners) were digitized directly into the database. About 30 percent of the mines were entered in this manner.

Some mine maps had no section corners, or only one. These mines were digitized along with any landmarks that could be used for orientation (north arrows, roads, railroad tracks, landforms, streams, or mine shafts). The mine outlines and landmarks were plotted at 1:24,000 and overlain on the appropriate USGS 7.5-minute quadrangle. Using the reference points and features digitized from the original map, the mine outline was registered to the topographic map and transferred by hand onto mylar overlays. Mines that were too narrow to digitize as polygons were drawn onto mylars as lines. Mine shaft and mine tunnel symbols found on quadrangles or other maps were also transferred to the mylars as point locations. The mylars were digitized after all mines for that quadrangle were compiled. About 50 percent of the mines were entered in this manner.

When the only information available for a mine was its legal description (township, range, section, quarter section or footages), X-Y coordinates were calculated from the legal description using a computer program and a database of section corner coordinates (Swann et al., 1970). The computed coordinates were entered directly into the non-coal mine database.

Documentation of Uncertainties

Documentation maintained for each mine includes date and source of the maps showing the mine outline or point location, and other sources of information. Possible errors in the source map or compilation process were also recorded. Every mine polygon, line and point entered into the database was assigned a code to indicate the accuracy of the source map and the method used to digitize or enter the data (Table 1). For example, polygons digitized from the original mine maps received a code of 1. Mine locations calculated from a legal description that located the point to the nearest quarter-quarter section were given a code of 8. These location-uncertainty codes were used during data synthesis to create buffer zones that covered the area where mines might be located.

SYNTHESIS AND EVALUATION OF DATA

The ARC/INFO GIS software is used to process the digital data (Morehouse, 1985). Areas at risk of subsidence damage are calculated by merging data from five spatial data sets and one tabular data set (Table 2). These data sets are physically large (76 Megabytes total), derived from source materials of different scales, and stored in different geographic subdivisions. The GIS provides a mechanism to 1) manage the data by extracting it in standard subunits of manageable size, 2) represent the proximity of certain features and account for uncertainties in boundary locations, 3) adjust and register spatial features from small-scale maps to features from large-scale maps, 4) merge spatial features and link to tabular data organized in different statistical areas, and 5) present complex information in a comprehensible manner.

Table 1. Location-uncertainty codes and buffer distances

Code	Buffer distance (ft.)		Source of mine outline or location
	proximity	uncertainty	
1	500	1000	Original mine map, four reference points
2	500	1000	Original mine map, registered using landmarks
3	500	1000	Topographic map
4	500	1000	Map with topography OR with scale larger than 1:24,000
5	500	2320	Map without topography AND scale smaller than 1:24,000
6	500	1000	Legal description with footages or good landmark
7	500	1660	Legal descriptions; section 1/4 1/4 1/4 or CE1/4 or CE1/2 1/4 or CE1/2 or CE1/2 1/2
8	500	2320	Legal description; section 1/4 1/4
9	500	3640	Legal description; section 1/4 or 1/2 1/4
10	500	6280	Legal description; section only

Table 2. Original scale, size and geographic subdivision of digital data sets

Data set	Scale of Source maps	Size of file (Mb)	Unit of storage
Coal mines	1:1200 - 1:62,500	11.0	county
Non-coal mines	1:1200 - 1:63,360	2.5	county
Land cover	1:250,000	15.3	USGS 1° x 2° quadrangle
Census tracts	1:62,500	10.0	SMSA*, county
Political townships	1:500,000	0.8	state
Census statistics	tabular data	36.0	state

*SMSA is Standard Metropolitan Statistical Area.

Management of Data

A number of complex processing steps are required to merge the data sets and to compile township-level statistics for mine subsidence potential. The full data sets are too large for the hardware and software to conveniently handle and too complex to for us to effectively monitor the results of incremental processing steps. To alleviate these problems, we process the data on a county basis. The county subsets of the main data sets are referred to here as coverages. The procedure for extracting the data for each county is transparent and relatively efficient with the use of map libraries.

A map library is a special data structure supported by the ARC/INFO software (Keegan and Aronson, 1985). Conceptually, the library consists of layers and tiles. The layers can be thought of as maps of individual data sets such as coal mines,

non-coal mines, and land cover. All layers are divided into the same set of geographic subdivisions called tiles. We use two libraries. The mine data library has counties for tiles and five layers: coal mine polygons, coal mine points, non-coal mine polygons, non-coal mine lines, and non-coal mine points. The second library has one layer, land cover, which is divided into tiles corresponding to 1- by 2-degree quadrangles. To process a county we create an outline of the county that extends one mile beyond the actual boundary, to allow for nearby urban land or mines. Given this outline, the GIS librarian software automatically determines which tiles are intersected in each library, retrieves the data from those tiles, and creates a single coverage for each layer.

Representation of Proximities and Uncertainties

Modeling to evaluate the exposure of structures at risk of subsidence must include consideration of the proximity of mines to urban areas and the uncertainties of the position of mine areas and urban boundaries. In this study, proximities and uncertainties are represented by buffer zones of various distances created around mines and urban areas.

Mine buffers. The GIS is used to create two buffer zones around mines. A small buffer (the proximity buffer) delineates the adjacent land that could be affected by subsidence; a larger buffer (the uncertainty buffer) represents the uncertainty in the position of the mine. Although the buffer distances for mines, if considered individually, would vary depending on the depth of the mine, nature of the geologic strata, quality of the available mine maps, and other criteria, the regional scope of this study makes it necessary to assign standard buffer distances to all mines.

Structures on land adjacent to a mine are at risk because the subsidence from the collapse of an underground mine can spread sideways as it moves upward to the surface. This lateral ground movement is not highly predictable, but is a function of the depth of the mine and the local geology, among other factors. Generalizing these factors statewide, we estimate that the maximum distance of lateral subsidence movement will be 500 feet, and use 500 feet as the proximity buffer distance on all mines.

Uncertainties in the positions of mine boundaries come from two sources: 1) incomplete or imprecise maps of mine workings and 2) errors in compilation and digitizing. We estimate that for all coal mines and many non-coal mines the error from these two sources generally would not exceed 1000 feet. This uncertainty is represented by creating a buffer zone extending 1000 feet beyond the proximity buffer zone.

The uncertainty buffer is expanded for mines located by small-scale maps and for mines with no map (Table 1). Because government regulations differ for non-coal mines, information on these mines is more difficult to obtain; maps of the workings are less likely to be kept on file and may not be available at all. When the legal description is the only source for the mine location, the uncertainty buffer distance is expanded according to the size of the area in which the mine might be located.

Urban buffers. A 1-mile buffer is created around all urban land areas (except for the transportation category). This buffer is to allow for uncertainties in the boundaries of the urban areas (some of them were based on mapping that occurred more than 10 years ago) and to identify areas where nearby mines will underlie future urban expansion.

Adjustment of Features

Before the spatial features for a county are overlain, township boundaries must be merged with census tract boundaries where they should coincide. Because the township lines and the census tract lines come from different source maps, there are slight offsets in some areas where the lines should be coincident. Using the GIS, township lines that are within a specified distance of tract lines are automatically shifted to match the tract lines.

Merge Spatial Features and Link to Tabular Data

After the extraction of the data into manageable county areas, the creation of buffer zones, and the adjustment of township lines to tracts, the coverages are merged into a single county coverage containing urban land, buffered urban land, buffered mined areas, townships, and census tracts. To calculate the number of homes at risk of mine subsidence damage, the merged spatial coverage must be linked to the tabular data set of census statistics. Also, the census statistics, which are organized on a tract basis, must be recomputed on a township basis for the final tabulation. The merged county coverage, still containing data on townships and tracts, provides a means for this calculation. The relational database management capability of the GIS is useful for linking and recomputing these elements.

The county coverage of census tract polygons is constructed from a DIME file of lines of tract boundaries. A program relates the lines in the DIME file to the census tract coverage and assigns tract numbers to the polygons. The polygons in the final merged coverage then have both a township number and a tract number. Each township may contain several tracts or sections of tracts. Another program uses the township and tract numbers to pick out the appropriate polygons for each statistical area, summarize and save the data for each township and for each tract into new files. Data on number and value of housing units per tract are stored in separate tabular files and are linked to the saved files. Through several relates of saved files and tabular files, statistics for each township are extracted.

Presentation of Data

The final step of this study is to produce and present statistics that can be used to evaluate the potential for damage from mine subsidence. For purposes of this study, undermined areas and areas within the proximity buffer are considered areas of highest risk. Areas within the uncertainty buffer are considered areas of moderate risk. Areas outside of both buffers have the lowest risk. Urban areas, particularly those classed as urban residential, are considered to have the highest concentrations of insurable structures. Areas within the urban buffer may have significant concentrations of insurable structures now or in the future. Areas outside of the urban buffer are assumed to have a low density of insurable structures.

Assumptions must be made about housing values and the distribution of housing units within a township. A comparison of figures from census tracts and census places in this county indicated that 80 to 90 percent of the housing units are in the residential areas. Therefore, for this example we assume 90 percent of the housing units to be in residential areas and the remaining 10 percent to be evenly distributed throughout the rest of the township. All housing units are assumed to have equal value. The GIS calculates the number and value of housing units undermined based

on the total number and value of units in the township and the percentage of residential land undermined.

Table 3 shows some of the statistics produced for a county in Illinois. Although less than one percent of the county is in the highest category of subsidence risk (i.e. land within the proximity buffer), more than 7 percent of the residential and other urban land is in this category. An additional 5 percent of residential land (4 percent of all urban land) is in the moderate risk category. In this particular county, 66 percent of the buffered mine area falls within the urban land and an additional 28 percent within the urban buffer. Six percent of the buffered mine area underlies water areas and does not present a risk of subsidence damage.

The data can also be effectively depicted in map form. Figure 1 shows most of the map features used for the evaluation of subsidence for this county; the tract boundaries and proximity buffers are not shown. The coincidence of mines with urban areas and urban buffers is readily apparent. Maps like this could be used to show the need for proper building practices and regional planning. Figure 2 shows the estimated number of housing units in the high and moderate zones of subsidence risk. Figure 3 shows the estimated percentage of housing units in the township that fall in these two zones. These graphics show the statistics of Table 3 in a spatial context, and present at a glance the geographical areas most at risk of mine subsidence damage.

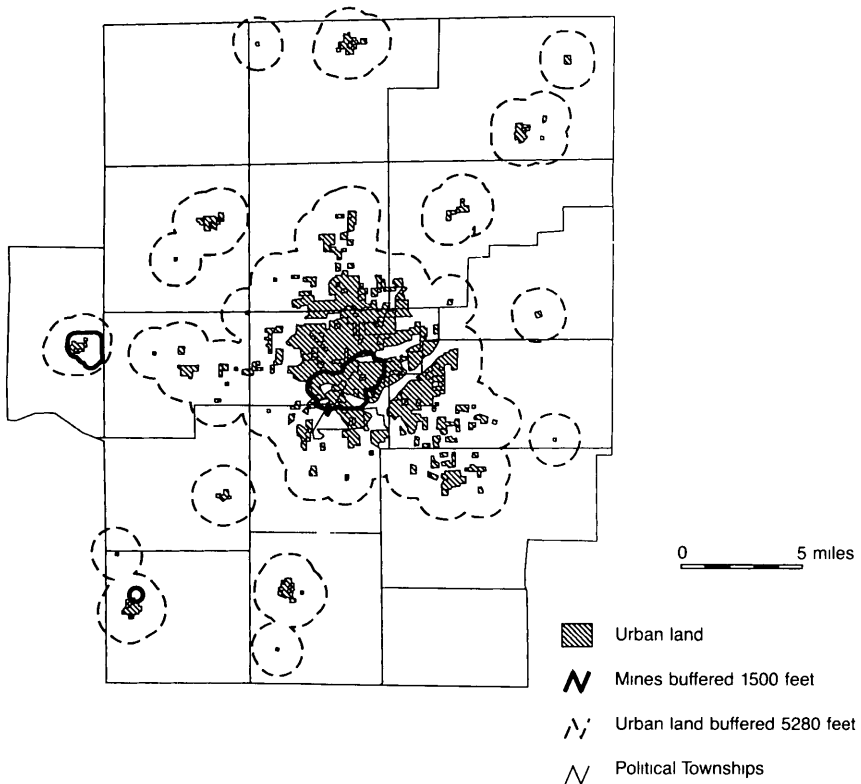


Figure 1 Merged map features

Table 3 Acreage exposed to high and moderate risk of mine subsidence

Twp. no.	Township			Residential			Other urban			Urban buffer			Housing units at risk	Housing value at risk
	total area	percent at risk	total area	area at risk	percent at risk	total area	area at risk	percent at risk	total area	area at risk	percent at risk			
High risk (undermined or within proximity buffer)														
8	18,406	568	3.1	135	94	69.7	52	100	3,324	422	12.7	225	\$5,398,505	
10	19,073	1,889	9.9	7,687	1,033	13.4	4,149	13.7	6,233	147	2.4	4,031	78,586,892	
13	17,262	0	0.0	888	0	0.0	339	0.0	8,584	0	0.0	0	0	
15	20,466	18	0.1	227	0	0.0	66	0.0	4,206	18	0.4	0	0	
Totals	371,444*	2,475	0.7	16,427*	1,127	6.9	7,946*	7.8	97,952*	587	6.0	4,256	\$83,985,397	
Moderate risk (within uncertainty buffer)														
8	18,406	518	2.8	135	40	29.6	52	0	3,324	479	14.4	96	\$2,275,166	
10	19,073	1,186	6.2	7,687	763	9.9	4,149	6.8	6,233	29	0.5	3,013	65,990,713	
13	17,262	7	0.0	888	4	0.5	339	0.0	8,584	3	0.0	6	304,822	
15	20,466	143	0.7	227	12	5.1	66	0	4,206	132	3.1	32	678,425	
Totals	371,444*	1,854	0.5	16,427*	819	5.0	7,946*	3.5	97,952*	643	0.7	3,147	\$69,249,126	

* Includes all other townships in county.

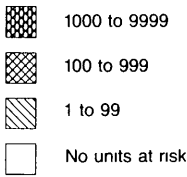
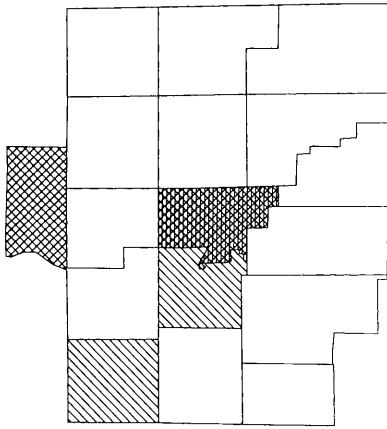


Figure 2 Number of housing units at risk by township

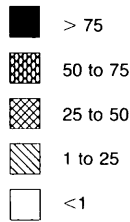
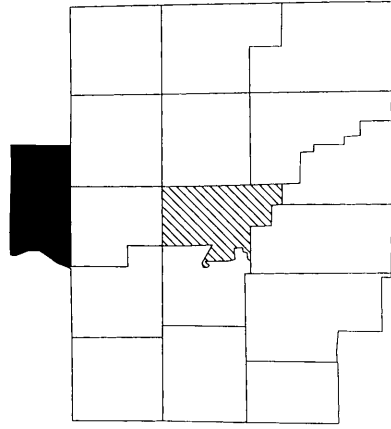


Figure 3 Percentage of housing units at risk by township

CONCLUSIONS

Use of a GIS is providing dramatic new views of the risk of damage to structures from mine subsidence. With GIS capabilities we can efficiently merge a number of large, diverse data sets and evaluate the risk of damage. Although it is up to IMSIF to act on the results of this study, this information will be useful for reviewing and modifying the insurance rate structure and the geographic areas of and procedures for marketing mine subsidence insurance.

The successful application of a GIS to mine subsidence risk could be expanded in several directions. As we learn more about the factors that contribute to subsidence, we can refine the categories of risk by mapping the factors and adding them to the merged county data sets. The GIS could be used to help identify these factors by finding spatial correlations between subsidence events and other parameters. Insurance companies may even be interested in using the address matching capabilities of the GIS to merge mine buffers, DIME (or TIGER) files, and customer lists to identify the homeowners who should be alerted to the need for mine subsidence insurance coverage.

REFERENCES

- Donnelley Marketing Information Services, 1986, Demographic Methodology, 13p.
- DuMontelle, P.B., S.C. Bradford, R.A. Bauer, M.M. Killey, 1981, "Mine Subsidence in Illinois: Facts for the Homeowner Considering Insurance", Environmental Geology Notes 99, Illinois State Geological Survey, 24p.
- Fegeas, R.G., R.W. Claire, S.C. Guptill, K.E. Anderson and C.A. Hallam, 1983, "Land Use and Land Cover Digital Data", U.S. Geological Survey Circular 895-E, 21p.
- Geographic Data Technology, Inc., 1982, "Tract-80, 1980 Census Tract Boundary Coordinate Files, File Documentation", 8p.
- Illinois Mine Subsidence Insurance Fund, 1987, Annual Report 1986, 12p.
- Keegan, H., and P. Aronson, 1985, "Considerations in the Design of a Digital Geographic Library", Proceedings of Auto-Carto VII, ASPRS, pp. 313-321.
- Loelkes Jr., G.L., G.E. Howard Jr., E.L. Schwertz Jr., P.D. Lampert, and S.W. Miller, 1983, "Land Use/Land Cover and Environmental Photointerpretation Keys", U.S. Geological Survey Bulletin 1600, 142p.
- Morehouse, S., 1985, "ARC/INFO: A Geo-relational Model for Spatial Information", Proceedings of Auto-Carto VII, ASPRS, pp. 388-397.
- Swann, D.H., P.B. Dumontelle, R.F. Mast, L.H. Van Dyke, 1970, "ILLIMAP-A Computer-Based Mapping System for Illinois", Illinois State Geological Survey Circular 451, 21p.
- Treworgy, C.G., M.H. Bargh, C.A. Hindman, C.A. Morgan, 1988, "Costs and Benefits of GIS Data Management: A Case Study of a Database Managed by a State Agency", Technical Papers of ACSM-ASPRS/1988, Vol. 2, pp. 186-195.
- U.S. Department of Commerce, Bureau of the Census, 1980, "Geographic Identification Code Scheme", 1980 Census of Population and Housing, p133-162.

DIGITAL DATA : THE FUTURE FOR ORDNANCE SURVEY
M SOWTON
ORDNANCE SURVEY
ROMSEY ROAD, MAYBUSH, SOUTHAMPTON, SO9 4DH

ABSTRACT

Ordnance Survey has been producing digital data at large scales (1:1250, 1:2500 and 1:10 000) since the late 1960's. Progress since then and the major factors affecting the development of digital mapping in Great Britain are described.

A number of issues are coming to a conclusion with the implementation of a pilot database, based on the successful study project and prototype trial which ran from 1985 until 1987. The paper explains the uses which will be made of topologically structured data for automated map production, customer designed maps and specialised datasets for various purposes.

The OS large scale digitising programme now includes blocks being digitised by the Utilities (water, gas, electricity and telephone). New data specifications and quality control procedures have been introduced to ensure acceptable data standards are achieved.

With the increased number of digital maps available increased capacity to revise the digital data is needed. Recent projects have confirmed that a field office can function satisfactorily without record maps.

INTRODUCTION

The Ordnance Survey of Great Britain (OS) is a Government Department with responsibility for the provision of topographic information in forms which customers require and can be economically supplied.

The original surveys of OS in 1790 concentrated on mapping at one inch to one mile, and later six inches to one mile, but modern mapping has been concerned with much larger scales. Today the official map coverage of Great Britain is as follows:

1:1250	54 600	km ² sheets - urban areas
1:2500	164 600	1 km ² sheets - developed rural areas
1:10 000	10 200	5 km x 5 km sheets - full cover derived from the above scales except in mountain and moorland areas
1:25 000	1 374	10 km x 20 km sheets
1:50 000	204	40 km x 40 km sheets (nominal size)
1:250 000	9	sheets
1:625 000	2	sheets

Not all these sheets have been digitised, but at present large scale digital data exists for:

1:1250	35 210 sheets (65% coverage)
1:2500	19 135 sheets (12% coverage)

Although various survey techniques have been used over the years to create these maps they are now kept up-to-date by predominantly graphic survey methods supported by instrumental survey and some photogrammetry. The up-to-date survey of an area is held in a local field office as an inked plot on a plastic field document known as the Master Survey Drawing (MSD). This method has proved to be extremely cost effective and digital revision methods have been designed to maintain this process as well as incorporate output from field instruments and photogrammetry.

Only in the last 2 - 3 years has interest in digital data caused OS to reconsider the nature of digital data as a product in its own right, and not purely as a technique of map production. Nonetheless the bulk of OS revenue is realised from the sale of large scale graphic products and royalties paid for copying and other use of these maps. Digital data is increasing in importance as a product, but for some years it is likely to be a lower revenue earner than graphics.

Graphic products are sold in 3 main forms. Firstly as printed copies which as time passes, become more and more out of date until a new edition is produced. Secondly as enlargements from 35mm microfilm output known as SIM (Survey Information on Microfilm). And thirdly as direct copies from the MSD produced in the survey office known as SUSI (Sale of Unpublished Survey Information). This allows the customer to have access to up-to-date mapping but has limitations regarding presentation. These services or improvements of them have to be maintained from the digital data as part of a fully viable digital process.

With this background in mind it is perhaps best to consider the development of the large scale digital processes at OS in 4 separate periods before finally taking a brief look at the future prospects.

Origins of OS digital mapping	1968 - 1972
Digitising Progress	1973 - 1982
Recent Past	1983 - 1987
Current Events	1988 - 1989

ORIGINS OF DIGITAL MAPPING AT OS

In the late 1960s a study of the possibilities of using digital techniques was carried out. At this time computer processes for automating engineering and architectural drawing processes were emerging, and it was considered that similar techniques could be adapted to the process of large scale map production.

Two approaches were studied. Initially production of digital data directly from photogrammetric plotting machines was investigated, but eventually the development of digital procedures centred on the digitising of existing maps. These early investigations concentrated on the reproduction of the original map to the same standards of accuracy and quality.

The reasons behind the adoption of digital processes were based on the benefits which were considered would arise in the automated reproduction of the maps and the derivation of smaller scale maps. The early data was collected as a series of strings captured in a sequence decided by the digitising operator. Each feature was assigned a feature code and a serial number, but the extent of each feature was determined more by digitising convenience than any other consideration. Data for each map was held in separate files on magnetic tape which started with a map header followed by the map data.

Feature codes were attached to the data in order to identify what each represented on the ground, and later to allow features to be suppressed in derived map production.

The digitising was blind, checked on a plot and corrected off-line, and this process was repeated until an acceptable result had been achieved.

DIGITISING PROGRESS 1973 - 1982

The processes described above continued to be refined with the introduction of more efficient data specifications and storage, interactive screens and improvements in techniques, but the basic concepts remained unchanged.

With only few customers for the data and conscious of the need to create a cost effective use for it, OS experimented to a greater degree on map production processes and the creation of derived maps from the digital data, rather than investigations into using it as a product in its own right. Despite all the effort expended, it was not possible to achieve 100% facsimile reproduction of the large scale maps because the available plotters could not cope with ornate symbolisation, a problem exacerbated by a data structure which did not support polygons. As a result much of the ornamentation had to be added by hand to a plot of the digital data. This partially accounts for the lack of success in deriving 1:10 000 scale maps from the larger scale data. Other contributing factors were an inappropriate data structure for computer generalisation, and a traditional specification on which 1:25 000 derived production depended.

Despite limited achievement during this period digital mapping was stimulated by the report of the Review Committee of OS, known as the Serpel Committee which firmly endorsed the policy of converting the large scale maps into digital data. Its recommendations were a great

support for the continuation of the digital programme.

As a result the potential of digital map data was investigated by a growing number of users, and in some areas trials of data were set up in collaboration with OS. The most significant of these trials which became known as the Dudley Project, was really two separate projects, one developing out of early structured data experiments and a second using the digital maps as a basis for an information exchange and record system for the Public Utilities. Both these trials were highly significant in the development of digital map use. In practice, the Utility project has had most impact because it focussed attention on the importance of using a common digital map base for the recording of Utility plant records and the need for such data to be provided as quickly as possible. The structured data project was more significant in terms of spatial analysis and the benefit which could accrue from such use, but unfortunately was overshadowed by the Utility project where the maps were only used as a background to the Utility plant records. The structured data project was undoubtedly the precursor of GIS in Great Britain, but was probably too advanced for the state of the art at that time.

OS had to design digital conversion and revision policies without having a definite user requirement. Clearly it was not sensible to produce data which no-one wanted and equally doubtful to maintain existing data which no-one had bought. However, it was decided to continue with digital conversion, a system to revise digital data based on existing graphic survey methods which became known as DFUS (Digital Field Update System) was being developed, and digital photogrammetry was revived.

In 1982 a House of Lords Select Committee on Science and Technology studied the subjects of Remote Sensing and Digital Mapping. Their recommendations included further support for digital mapping and the acceleration of the OS Digitising Programme.

Thus by the end of 1982 there appeared to be a growing interest in the digital map data itself which was available for about 25% of the urban areas. The need for digital mapping had been established through the Review Committee of OS, the Select Committee of the House of Lords and OS had itself held seminars to establish the needs of users. Nonetheless it was still largely an interest expressed by potential users rather than a commitment to purchase data and there were few indications about where it would be most worthwhile to produce the additional data. The ability to update the data and processes to derive smaller scale maps were being investigated and emphasis was increasing on ways to accelerate the digital conversion.

RECENT PAST 1983 - 1987

This was the most active period in the development of digital mapping in Great Britain. After the Government response to the report of the House of Lords Select Committee the OS made representations to the Treasury for

additional funds to accelerate the digitising of the large scale maps. These funds were used to employ outside contractors on the digitising process.

Up to 1985 the digital data was stored in serial file form in a tape databank, but it was becoming clear that some improvement was essential. The opportunity was taken to launch a study into the requirements for digital data by both customers and the OS itself together with a review of the techniques of storage, handling and management. The study team carrying out this project produced its report in December 1987, having implemented a prototype database based on relational technology using a small Briton Lee database machine.

While the database study proceeded the improvement of DFUS and the investigations to integrate digital data from photogrammetry and instrumental survey continued and for operational reasons the tape databank was replaced by magnetic disc storage but still in serial file form.

As a direct outcome of the House of Lords Select Committee, in 1985 the Government set up a Committee of Enquiry into GIS under the chairmanship of Lord Chorley. The enquiry added considerably to the already growing pressure on OS to increase the output of digital map data, and within its recommendations were important proposals for OS to collaborate with its major customers in funding and acceleration of the digital conversion programme. Subsequent high level negotiations with the Utilities led to an agreement whereby customers for digital data in areas not yet digitised would let contracts for digitising within the framework of an OS controlled programme. This digitising would be carried out to a reduced feature code specification known as OS 1988, and a quality control procedure based on statistical sampling was introduced. Data which met the specification would be accepted into the National Topographic Database, kept up-to-date by OS, and sold to other customers to the mutual benefit of OS and the customer letting the contract. Thus, completion of the digitising programme became a realisable objective for the near future.

In this period a trial was started to investigate the possibility of using digital data to support a field survey office, which did not hold any map graphics, but which could satisfy its needs for all survey drawings directly from the digital data. The trial area was established in Milton Keynes where collaboration between the OS Agent, the Development Corporation, local customers and the OS field office tested the concept of a "map-less" field office. In addition to this primary aim Project 88, as it was called, was expected among other things to investigate the practical and economic limitations of supplying a wide range of high quality graphic products directly to the public from the field office. The equipment for the trial, which was a variation of the standard DFUS, was installed in July 1987 and the system was worked up in parallel to the conventional revision methods so that by the launch date in

January 1988 all the MSDs for the trial area could be destroyed.

At the end of 1987 the team carrying out the database study reported favourably about the introduction of structured data with predictions of cost savings in the production of graphic maps, and also produced an implementation plan for a pilot project to test their conclusions. The study concluded that significant benefits would accrue from:

- a completely digital large scale map archive
- topologically structured data
- the addition of management, quality and process data to the topographic data

and that as a result of these changes in data specification, there would be additional benefits if database management were to replace file management, and more on-line communications were to be introduced. The implementation plan for the Pilot Topographic Database, as it was called, contained 3 main elements:

- to develop a fully automated map production system which would eliminate all manual intervention
- to create a pilot area to test the conclusions of the database study
- to carry out Marketing and R&D projects to examine the implications which would arise from the full implementation of a data management system based on the conclusions of the database study.

CURRENT EVENTS 1988 - 1989

The early part of 1988 was one of consolidation, the implementation plan for the Pilot Topographic Database was accepted, Project 88 was launched, the feature code specification OS 1988 and the quality control procedures were finally agreed, agreement was reached with British Telecom and other Utilities over the terms on which they would digitise areas where digital mapping could not be produced in time by OS, and contracts for the supply of digital update by OS at various levels of change were arranged with a number of customers. The possibility of a completely digital large scale operation which was cost effective was beginning to emerge.

Project 88

The outcome of this project has been the acceptance of the financial and technical advantages of using a field system to maintain the digital data. The ability to discard the graphic survey records (MSDs) and rely completely upon the digital data to furnish the day to day needs for field documents has improved efficiency, increased output and reduced the cost of the equipment. Authority has been given to deploy 12 additional field update systems currently based on SUN workstations and a number of offices are being converted to "map-less" working.

A fundamental addition to the OS product range was also possible with Project 88. In the past customers could buy SUSI or SIM as the most up-to-date survey information, but with up-to-date digital data it was possible to provide customers with high quality graphics to a variety of specifications, scale and sheet layout directly from the field office.

Instrumental Surveying and Photogrammetry

While Project 88 concentrated on the main issue of how to automate a very economic, but low order of survey, other projects have concentrated on how to merge data from higher accuracy survey methods into the data. Techniques to integrate the coordinates of points fixed by Instrumental

Survey into the database have been developed. Data recorded by survey instruments is now coded and incorporated directly into the digital archive, immediately after capture.

Similarly with photogrammetry, as a result of developing new techniques there is now less movement of detail at the field completion stage and most digital data from photogrammetry goes into the database unaltered.

The ability to assign quality codes to each of these data sources to show that it is better than data originating from cartographic digitising or graphic survey is being developed as part of the Topographic Information System (TIS). (See later)

Pilot Topographic Database

The Pilot Topographic Database has now been publicly launched, although not all areas of the pilot can go live before April 1989. The original intention to select one area was not possible and a compromise has been adopted to create a database covering more than one area where different aspects of the project can be tested.

The main considerations in the pilot project are:

- Relational database management for the topographic data with improvements in data handling and storage of information related to the data.
- Structured data with object building and attribute attachment.
- Increased production efficiency with the eventual establishment of a Topographic Information System.

Relational Database Management

In the earlier study a relational database was identified as the most efficient way in which structured data could be stored together with the associated objects and attributes. A software relational database management system was tested on a standard computer where it was found it be very slow, and a much larger computer would be necessary to give a

realistic performance. Tests on a specialist database machine from Britton Lee showed this type of machine to be more suitable for small amounts of data.

Government procedures for the purchase of computers is slow and although OS could have called for tenders on a given operational requirement, this would have delayed the start of the pilot project. It was eventually agreed that OS should purchase a larger Britton Lee for the pilot project and seek competitive tenders for suitable computer equipment before the implementation of the full system if the outcome of the project endorsed relational database management with structured data as the best way forward. A Britton Lee IDM 700 model 140 was obtained for the pilot project.

Structured Data

The bulk of OS digital data remains much the same as that described earlier. The structured data is produced from

this by digitising some additional information, mainly polygon points, creating a link and node structure by software with some interactive editing and building objects of the polygon type also by software, using the polygon points which can then be discarded. The attributes of these polygons can then be used to fill areas with symbols, other ornamentation or colour. The topological relationships thus identified are explicitly stored in the database along with attributes deduced from the original data. Positioned text, previously used for cartographic purposes is now being attached to the points, links, nodes and objects to which it actually belongs as an attribute.

Topographic Information System

Although the full details of the TIS have not yet been developed, the concept is simple. Management data will be held as relationships or attributes in the database associated with the topographic data, allowing information about production control, revision status, data quality, history (date of survey or demolition), rates of development, attributes defining treatment of data at other scales etc to be recorded for objects or features within the database.

It is anticipated that the Topographic Database will be linked with other databases associated with the graphic maps such as the Map Information Database and the Digital Marketing Database to create a comprehensive central system for topographic information, that is, the TIS.

The Pilot Project Area

In order to test one of the main benefits identified in the earlier study, it has been decided that all new editions since the start of the project will become part of the pilot database project. This will demonstrate the cost advantages which are expected from the production of new editions directly from the structured data. This decision means that

a number of isolated maps all over the country will be created in structured data to test this map production process.

Three areas have been selected to test other applications. Most important is a block of data covering Tameside Metropolitan Borough in Greater Manchester where the Borough is committed to a trial of GIS and is also prepared to produce maps from the data for OS customers. Of slightly less importance is a small block of maps wanted for a pilot trial of GIS for the City of Birmingham, which together with a larger block also in Birmingham, makes up the second area. The combined block could be used to extend the pilot GIS if successful, but is needed to prove that an OS map agent can use the digital data and produce graphics for customers in his own premises. Finally, there will be a block of mapping covering the Project 88 area in Milton Keynes where it will be possible to test the relationship between the structured data and the Project 88 operation.

It is anticipated that the structured data produced in these 3 areas, together with the structured data for new editions will allow OS to evaluate the advantages arising from it, and the use of a relational database in a production environment. The significant customer linked advantages which need to be tested are:

- The use of structured data outside OS.
- The ability to produce graphic products directly from the data.
- The creation of additional digital products.
- The production of microfilm output.
- The handling of digital data by agents.

Graphic Products from the Data

For some time it has been recognised that digital data will only be a valuable asset for OS if it can be harnessed to produce not only a range of digital products, but also a range of graphic products similar to the current range of printed paper maps and copies of the surveyors drawings like SUSI and SIM.

It was realised that this could only be done by eliminating manual cartographic processes, by substituting software processes to create similar effects, and by modifying the map specification within acceptable limits. A flowline has been created so that in future, all new editions will be produced in this way. This process has become known as AMP (Automated Map Production) and its success depends on structured data, some additional feature codes and a high quality electrostatic plotter. It has at last been demonstrated that digital production of maps is cheaper than could possibly be achieved by conventional manual methods. As an extension of this process it is possible to carry out a variety of manipulations on the data:

- Altering the sheet layout.
- Changing scale.

- Suppressing detail.
- Adding symbols.
- Adding colours.

All of which can be achieved through the medium of the structured data. A customer wishing to have a non-standard product to his own layout, scale and specification can within limits, specify what he needs and have a plot within a short period. This service has become known as CPCD (Customer Plots from Current Data). Both AMP and CPCD depend on the map data being up-to-date to achieve their maximum potential. If this can be done on a regular basis then the current SUSI service would be replaced by an up-to-date high quality graphic output.

Now that all the processes have been perfected AMP is capable of producing standard 1:1250 new editions where digital data exists, giving a result little different from printed new editions on chart paper. A recent breakthrough in the way in which areas are calculated and parcels are numbered has made it possible for 1:2500 new editions to be produced in the same way.

Location of Map Production Facilities

The map production processes described above are very fast. With up-to-date digital data it is clearly not sensible to print maps and store them pending the arrival of a customer. The sales and distribution processes are being reviewed and experiments are in hand to place the map production outlets throughout the country. Using local storage of data or communication links, OS is installing plotters and processors in a survey office, map processors sales agent and a Local Authority to test the viability of a "walk in and buy" map service.

Additional Digital Products

With structured digital data it is possible to create datasets for specialised uses. As part of the pilot database trial, three such datasets will be made available to the public.

- OSBASE: The standard structured digital map data usable as a base for GIS and graphic plot production.
- OSLAND: A product based on OSBASE with all the land and highway parcel polygons closed and referenced. This dataset includes additional feature codes, reference numbers, post codes and postal addresses.
- OSCAR: A dataset related to the road network for use in navigation systems and for solving road related management and maintenance problems. This dataset includes road names and classifications linked to each network segment.

It will be possible for the OSBASE dataset to be delivered to customers indistinguishable from current data, since the advantages inherent in the data for OS map production processes can be suppressed for those users not wishing to exploit the benefits of structured data.

Microfilm Output

It is possible to produce microfilm directly from the existing digital data, but because that data does not include all the details required to draw a complete new edition, microfilm produced in this way is incomplete. However, using structured data in a variation of the AMP flowline it will be possible to produce an up-to-date microfilm as soon as significant change has been recorded.

It is considered that this will remain a standard outlet for up-to-date graphic output for a wide range of customer for some years to come, particularly in areas where change is small and digital production methods by OS agents are not justified.

Enlarged prints from microfilm produced in this way would be indistinguishable from the AMP product and would have the advantage of being frequently updated unlike the current SIM.

FUTURE PROSPECTS

Production of digital data has increased rapidly in recent years. First as a result of the reduction in the feature code specification, secondly by the employment of contractors by OS, and thirdly through data produced by contracts let by the Utilities. This has increased the total amount of digital data and consequently the sales of the data, but has also placed greater emphasis on the requirement to keep the digital data up-to-date and to manage the data.

The results of Project 88 have shown that a cost effective solution for the revision process based upon digital data is possible, and additional equipment will be deployed in step with the requirements to maintain digital mapping in areas of high demand.

If the technical benefits of structured data are proved to be cost effective by a successful outcome of this part of the Pilot Topographic Database project, AMP and CPCD can be introduced as standard map production processes from which significant savings and opportunities to increase sales of large scale graphics should result.

The Pilot Topographic Database is due to report towards the end of 1989. The prime issues will be the use of structured data outside OS and how such data is to be created, manipulated, revised and stored. Whether a relational database is essential for the efficient handling of structured data and the creation of a Topographic Information System will form an important part of the results. Thus the success of current trials of GIS will

have an important bearing on the extension of structured data into other areas.

The introduction of structured data will create problems with the editing process which has not been designed to cope with topological relationships in the data. Investigations are in hand to create an Advanced Edit System to take account of this requirement.

Investigations into the possibility of using attributes rather than feature codes to derive 1:10 000 scale maps from the large scale data appear promising, and will provide additional justification for the introduction of structured data.

A major factor in the future will be the marketing of both the structured data and graphic products produced from it. Part of the Pilot Topographic Database project is concerned with customer requirements and the way they are met. The CPCD option lends itself to a direct output of a plot from digital data to a customer specification directly from the OS Agent, the field update system allows data to be plotted for customers in OS field offices, and AMP would provide up-to-date plots from Headquarters. Added to this are the various ways in which the digital data can be provided to fit different applications.

In areas where digital data exists the production of OS large scale mapping is now entirely digital. The uncertainties which remain should soon be resolved, digital coverage increased, and a digital future for large scale products ensured to the benefit of both customers and OS.

GIS, AM/FM, AND AUTOMATED CARTOGRAPHY IN JAPAN

Dr. Sachio Kubo
Ochanomizu University
2-1-1 Otsuka, Bunkyo-Ku, Tokyo, Japan

ABSTRACT

In these five years, introduction of GIS, AM/FM and automated cartography has been increased rapidly in Japan. AM/FM is leading the trend. Very large systems are operated by utilities and a central organization was set up under the control of the national government. CD-ROM data base and PC based GIS is becoming popular. Automobile navigation systems are going to take off. Institutional problems between and within agencies became an obstacle in popularization of new technology.

INTRODUCTION

Although Japan is the third largest country in the world in terms of numbers of users in GIS, AM/FM, and automated cartography, the situation in Japan is not known in English speaking countries. A survey in 1988 September issue of PIXEL magazine figures that there are 31 GIS, AM/FM, and digital mapping systems in Japanese market from 29 companies. This number does not include PC based systems. Some are from US (Intergraph, ARC/INFO, Synercom, McDonnell Douglas, and Computer Vision); one from UK (Laser Scan); and others are Japanese made systems. In 1984, a survey by NICOGRAPH shows there were no more than a dozen systems available in market. This figure shows that Japanese market is rapidly growing. However, under this situation, several problems are found. Compared to widely used AM/FM, GIS attract small users in Japan. Lacking synthetic national policy in spatial information systems, institutional problems became severer.

GIS

National agencies

Although GIS has been attracting interests of national agencies for two decades, very few GIS are used actually. Japanese government has been interested in building a nationwide data base for her land, and the National Grid Data, consists of land use, landform, DTM, soil, geology, transportation, population, economic activities, meteorological data and many other items, was built with several billion yen and twenty years. Unfortunately, this valuable data base is not fully utilized by agencies.

There are two major reasons for small amount of users; (1) access to data is limited to government, university and utility, and (2) agencies do not have tools to use data base. A typical case is found in the Geographical Survey Institute (GSI). Although GSI is the major provider of grid data, GSI does not have GIS. Almost the same situation is seen in the Ministry of Agriculture, Forestry and

Fishery (MAFF), who provides grid data on agricultural census, and the Statistical Bureau providing population census data. The only national agency using GIS for manipulating the National Grid Data is the Land Agency. ISLAND, a Fujitsu made mainframe based GIS has very poor graphic interface and is almost inutile. There are some plans to improve the situation. The Statistical Bureau will introduce ARC/INFO or similar in fiscal year 1989. Kokusai Aerial Survey Co. contracted to build PC-based GIS for MAFF.

Local Governments

Prefectures Prefectures and municipalities are largest GIS users. Saitama, Kanagawa, Chiba, Hyogo and Okinawa prefectural governments have introduced ARC/INFO for regional planning. Planning departments in Tokyo, Osaka, Hokkaido governments have decided to introduce GIS. It is forecasted that more prefectural governments will introduce GIS in 1989.

UIS Very early GIS users in municipalities are Kitakyushu and Nishinomiya, who received grants-in-aid from the Ministry of Construction in the early 1980s. In mid-1980s, several cities introduced GIS made by Japanese mainframe computer companies as "showcases". Numazu, where Fujitsu's research laboratory and factory are located, introduced Fujitsu made GIS "ARISTOWN", and Abiko, where NEC has a PC assembly factory, introduced "WING". In 1986, the Urban Department of the Ministry of Construction started a project UIS II, which promote municipalities to introduce integrated GIS for planning and administration. Koshigaya, Tokorozawa, Ogaki, Okayama and several more mid-sized cities are selected as the test fields.

Institutional problems Generally speaking, it is very difficult to build an single integrated GIS in very large cities. Yokohama, Kawasaki, Nagoya and Osaka are trying hard to build multi-purpose integrated GIS. Obstacles in building an integrated GIS are institutional problems between departments in local governments and between departments in national governments. Japanese administration system is more centralized than those of north American countries, thus the central government is influential in introducing GIS. One of the example is a conflict between the Housing Department and the Urban Department in the Ministry of Construction. The Department of Housing issued a notice that housing construction permit data should not be used in UIS II. Another conflict is between the Ministry of Construction and the Ministry of Home Affairs.

Emergency information systems One of recently emerging applications of GIS is emergency information system. The Fire Department of Nagoya City introduced Fujitsu's system in 1987. The place of fire or ambulance request is identified by address and large scale maps displayed in CRT. Explosive materials and homes of assistance required persons are also displayed. The system automatically makes up a fire extinguish plan, and send it to every fire station by facsimile. A copy is also sent to commander vehicles by radio facsimile. This type of system is being introduced into several cities including Osaka. The River Department of the Ministry of Construction set up the River

Information Center, from where local rainfall data and the water level of rivers are sent to local governments to prevent flooding.

Private industries

Utilization of mini-computer based GIS in private industry is limited to consultants, general constructors and surveyors with a few exception in forestry business. Recent trend is emerging utilization of PC based GIS. Zenrin, a map publisher in Kyushu, started distributing CD-ROM based large scale digital data with every individual household in 1987. Z-map includes a CD-ROM, basic handling software, a GIS construction kit, and a sample program. Equivalent products are in the market from several vendors including NTT. Stellar Co., a small software house in Tokyo, sells PC and EWS based GIS construction kits. Lion, one of the largest soap manufacturing companies in Japan, has formed a network of POS registers, and process sales data using a system based on Stellar's kit. MAFF's system, which was described previously, is also based on Stellar's kit. Users of PC based systems are increasing in banks, insurance firms and super market chains.

Research organization and universities

Only a few sets of GIS have been used in research organizations (e.g. The Geological Survey Institute) and universities. In Japan, cartography and surveying have not been concerned much in university. There is not even a single chair for cartography in Japanese universities! Recent recognition of importance of GIS education and research in geographers and planners resulted to reorganize university curriculum in several universities.

AM/FM

Background

Utilities now seems enthusiastic in introducing AM/FM systems. One of the strong motivations is strong yen and OPEC. Electric power and Gas companies now can buy oil less than the half price which they paid several years ago. Three quarters of the extra benefit from cheap oil is paid back to their customers, but a quarter is left to companies for new investments such as new plants, building utility conduit, and information systems.

Gas

Tokyo Gas Gas industry is a pioneer in introducing AM/FM in Japanese utility industries. Tokyo Gas, the largest gas company in Japan which owns total length of 36,000km pipes, serving to nearly seven million customers in the Tokyo Metropolis and the suburbs, has been building AM/FM system named Total Utility Mapping System (TUMSY). The service area of Tokyo Gas is covered by 27,000 sheets of 1:500 scale maps. Tokyo Gas started building the system in 1977. In the first stage, Tokyo Gas searched U.S.A. built systems, but found the existing systems are not suitable for the requirements, especially in processing kanji, then decided to build a system by itself. After seven years of development, the system was completed in 1983. TUMSY is operated on several VAX CPUs networked by

a VAX Cluster system. All CPUs are located in the headquarters in downtown Tokyo. Terminals are placed in the headquarters and several satellite offices where workmen can pull down the newest map or they can input altered information of piping. The system was recently improved to enable interchanging data to IBM mainframes where customer information including user family or corporate data, monthly gas consumption is stored. With this capability, TUMSY can be used as an integrated corporate information system. Tokyo Gas has sold TUMSY to nine gas and water utilities including Hokkaido Gas, Seibu Gas, Shizuoka Gas, and the Water Department of the Tokyo Metropolitan Government.

Osaka Gas The second largest gas company, Osaka Gas, is also developed a system named "IIS-MAP". Osaka Gas serves gas to Osaka, Kyoto, Kobe and their suburbs. The company started to develop a AM/FM system in 1974, but after three years the development was suspended. The problems were in the performance of computers and costs. The development restarted in 1983, and in the same year, the company decided to build a pilot system. In 1984, 471 map sheets are digitized, and the evaluation of the system was finished in December 1985. Building of an operational system was started in 1986. The system is consisted of an IBM mainframe, located at the central computing center, and distributed six IBM 9370 computers at branch offices. Each branch has about ten workstations. The software is a joint product of OG Information System (a subsidiary of Osaka Gas) and IBM. The software was sold to five users. These two large gas companies are becoming providers of AM/FM systems to smaller gas companies.

Electric power

In Japan, in the World War II time, a single electric power company was formed, and after the war, it was divided to nine regional companies. Among them, Tokyo Electric Power (TEPCO) is the largest private electric power generator and supplier in the world, who has approximately 20 million users. TEPCO has been developing a AM/FM system with Toyo Information System for managing half million maps. TEPCO also developed an automatic scanning system with Mitsubishi Electric. This AI based digitizing system can recognize crossing lines and eliminated objects in underpasses. IBM and Fujitsu are major distributors of AM/FM systems to electric power industry. Toshiba, Meidensha and Meitech also developed AM/FM for the same purpose.

Telecommunication

Nippon Telephone and Telegram (NTT), which was a part of national government until 1986, became one of the largest private companies in Japan. NTT developed several series of spatially referenced information systems. In AM/FM market, NTT is distributing "INS-SPACER" through INS Engineering, a joint company with Mitui Ship Building. NTT also developed a semi-automatic scanner "INS-CHASER".

Water supply and sewage

Water supplying and sewage service belongs mostly to municipalities in Japan. Compared to privately-owned energy and telecommunication service industries,

introduction of AM/FM is somewhat slower in government owned services. Tokyo, one of the largest cities in the world, populated more than 10 million, has been testing AM/FM for sewage service, and recently purchased a system from Tokyo Gas for water supplying service. Yokosuka City purchased a system from Fuji Electric. Sapporo, Osaka, Nagoya, Hiroshima, Okayama and other major cities also decided to introduce AM/FM. These cities are now in process of selecting systems or in performance evaluation stage. Before 1980, water supply services are in financial difficulties mainly from construction of new dams, waterways, and water reservoirs for increasing water demand. But precepts from severe water shortage in Kyushu Island and publicity for saving water thereafter decreased water consumption. At the same time, bursting of old mains laid a century ago became frequent. Cave-in in the main street of Ginza caused from leaking mains became head line news of TV. Maintenance of facility became a critical issue in the late 1980s. Almost the same situation is in sewage service. In the 1970s, rapid suburban development and river water pollution problem caused by septic tanks requested extension of piping networks. In 1980s, improvement of older systems in inner city and maintenance became more important. AM/FM for water supply and sewage service is forecasted as a promised market because there are more than 3,000 municipalities. A bunch of vendors are providing systems; they are Fujitsu, Hitachi, Intergraph, Fuji Electric, Tokyo Gas, Osaka Gas, Computer Vision, and value added resale agents such as Kubota Iron Works and Nippon Steel Pipe.

ROADIC

According to increasing utilization of AM/FM in utilities, exchange of data became essential. In 1985, the Department of Road in the Ministry of Construction set up a new organization "Road Administration Information Centre (ROADIC)". This semi-government organization aims to maintain data bases of roads and utilities in eleven cities with over million population. The center converts and updates road data base under contracts with road departments of national and local governments. Utility enterprises provide piping and wiring data to the center, and in return, they can access to both road data base and utility data base, so that they can exchange data base with governments and other utility enterprises. Governments can access data bases in order to manage roads as well as to charge utilities for using roads. Construction of a pilot system was started in July 1987 at their Kanagawa Branch (where maintains data bases for Yokohama and Kawasaki) using NTT's INS SPACER. In fiscal year 1989, the system at Kanagawa Branch will start operation. In other cities, operation will be started in fiscal year 1990. It is already decided that in Tokyo Branch, TUMSY will be used, and in Osaka, IIS-MAP will be used in ROADIC. There is a future plan to use this data base for city planning, property assessment and other purposes, but at the present moment, this plan is difficult to carry out because of institutional problems between and within ministries.

AUTOMATED CARTOGRAPHY

GSI

Two major projects have executed by the leadership of GSI. The first project is to establish a standard procedure and formats in digital cartography. This standard is limited to computerized photogrammetry and lacks viewpoints in future data production and GIS development. Another project is building a system called "Computer Aided Cartographic Processing System (CCPS)". CCPS is a EWS based map editing and printing system for 1:25,000 topographic maps. GSI plans to digitize all 1:25,000 maps within several years. When GSI started a project of digitizing 1:2500 scale maps in 1986, managers in GSI forecasted that sales of digitized maps would be increasing. GSI formed a semi-government organization for management and sales, and the organization requested surveying companies to digitize 1:2500 maps of Nagoya area by their costs. The data base was completed in early 1988, but GSI could find no user after all. Utilities rejected to use 1:2500 scale data because they need more precise maps for underground pipe description, Nagoya City evaluated that any data base without updated are not usable, and found digital map used by their fire department has better quality because it is updated every two months.

Marine Safety Agency

Two activities are pointed at the Marine Safety Agency. The first activity is a discussion on standards for electronic charts. Computerization of ships has been accelerated in these years and already numbers of ships are equipped with plotters, which plot the traces of navigation on CRT and recording devices. But at this moment, there is no standard in digital navigation charts. Determination of an international standard for electronic chart was requested by IMO/IHO. Another activity is building a data base of coastal zones along Japanese islands.

AUTOMOBILE NAVIGATION SYSTEMS

Toyota Crown, a flagship in small car line of big Toyota, received a full model change in 1987, and the top model was equipped with a navigator. Although there are still a bunch of arguments whether an automobile navigation system can be helpful or not, Japanese industry and government agencies are moving fast for realization. There are major two opposing groups; AMTECS, led by the National Police Agency, and a group organized by the Ministry of Construction. The National Police Agency and the Ministry of Construction have been argued for over twenty years on traffic control and guidance on highways. The basic concepts in two groups are identical. The location of an automobile is identified by XY coordinate information from the sign posts and sensors equipped with the automobile. Sign posts also send real-time information on traffic condition and parking facilities. Digitized map is stored in a CD-ROM with auxiliary information (e.g. amusement, restaurant). A CRT display shows a trace of vehicle on a map, and a voice synthesizer gives turning information and warning. GPS will be used auxiliary, because in urban

areas, the radio wave of GPS is difficult to catch in urban areas. It is not certain whether automobile navigation systems will be accepted by Japanese drivers or not, but it is obvious automobile industry likes it because they can mark higher price tags on new cars.

STANDARD

A committee to discuss on determining Japanese standards on digital cartographic data was organized in 1987. The members are from national agencies, local governments, utilities, surveying companies and a university. The committee published two reports in 1988 concerning basic ideas and activities in the United States. GSI refused to send members to the committee, and interfered the activity of the committee. The reason of opposition is that the committee is organized by the Standard Division of the Ministry of International Trade and Industry (MITI). GSI's refusal weakened the activity of the committee.

LEGAL ISSUES

Problems of data security and copyrights became a critical issue in public opinion. Over 400 communities have security acts and more than 700 have rules. In some communities, it is stated in act that collected data for a specific purpose should not be used for other purposes. Strictly reading the statement, a planning map cannot be used for tax assessment, or resident register cannot be used for making population statistics. This would be a future obstacle in using integrated GIS in local governments.

The focal point of the copyright problem is that whether GSI can claim copyright on digital maps. GSI only publishes small and mid scale maps, where large scale maps are mostly published by local governments. In Paper maps, when a publisher reproduces or use GSI maps for a base map, he should ask GSI for permission. But in digitizing a GSI map, does one have to ask permission? It is still unclear. In large scale maps, copyright problem is more complex. GSI claims for partial copyright for using GSI's public survey data. And in many cases, it is not clear that whether copyright belongs to local governments or to surveying companies.

THE NATIONAL MUSEUM OF CARTOGRAPHY

In April 21, 1988, the Science Council of Japan Issued a recommendation to the Prime Minister of Japan, Mr. Noboru Takeshita, on establishment of the National Museum of Cartography. The recommendation insists that in the era of globalization, it is urgent and national requirement to have a center to collect, store, process retrieve, and provide domestic and international geographic information including maps, charts, atlases, aerial photographs, ground landscape photographs, satellite images and various statistics.

Although the recommendation carries the name of museum, it is more than a display place of old maps. The museum will have five major functions; (1) map library, (2) research

center on geographic information and spatial analysis, (3) map exhibition and social education, (4) data base service, and (5) training and graduate level education. The recommendation is being processed by the National Council of Science and Technology. Nine ministries and national agencies have expressed their interests in establishment of the museum. Also several prefectures and cities are inviting the facility. The main hurdle is a strict policy of reducing government employees kept by the Japanese Government.

REFERENCES

- Kubo, S, 1988, ed. Computer Mapping, Nihon Keizai Shinbunsha, Tokyo
- Murai, S. 1986, ed. Proceedings AUTOCARTO JAPAN 2, Autocarto Japan Organizing Committee, Tokyo
- Kubo, S. 1987, ed. Proceedings AUTOCARTO JAPAN 3, Autocarto Japan Organizing Committee, Tokyo
- Kubo, S. 1988, ed. Proceedings AUTOCARTO JAPAN 4, Autocarto Japan Organizing Committee, Tokyo
- Kawauchi, 1988, ed. Recent Computer Mapping Systems, PIXEL, Gazou Joho Shori Center, Tokyo.
- Land Agency, 1986, ed. Geographic Information System, The Printing Bureau, The Ministry of Finance, Tokyo.

TRENDS IN COMPUTER-ASSISTED CARTOGRAPHY IN HUNGARY AND EASTERN EUROPE

Pál Divényi

Institute of Geodesy, Cartography and Remote Sensing
P.O. Box 546, 1373 Budapest, Hungary

INTRODUCTION

The situation in Eastern Europe on the field of computer-assisted cartography has so far been different from that in advanced countries of Western Europe. The activity in the theoretical domain is very extensive. Technical equipment of computer-assisted cartography and its spreading of use for geographical and cartographical systems in the society is, however, rather low at present. The main reason for this situation is a non-adequate level of hardware, lack of specialized equipment for displaying graphics and for processing image information.

After this short review I should like to introduce to the conference the current state and trends of computer-assisted cartography and GIS in Hungary and Eastern Europe ending with the activity of a quasi committee of COMECON on the field of computer-assisted cartography. The paper describes the Hungarian situation and selected information from Czechoslovakia, Poland and the USSR.

COMPUTER-ASSISTED CARTOGRAPHY AND GIS IN HUNGARY AND SOME EASTERN EUROPEAN COUNTRIES

Trends in Hungary

The activity of the community of Hungarian scientists involved in research and development in computer-assisted cartography covers a wide range of fields in accordance with the distribution of interest and funding.

As regards the data acquisition system, different interactive vector digitizing systems can only be mentioned, where software has been worked out in house. In addition, some experimental studies are under way related to automatic data collection, such as raster scanning maps and digitizing of stereomodels. Unfortunately, up-to-now there are not any line following or automatic digitizing systems for large format in Hungary.

Preliminary results of research concerning interactive data editing can be attributed, for the most part, to engineers due to the influence of CAD/CAM. In the mapping were born some useful results with a GRADIS system in a DEC environment and in PC type IBM AT/XT with own software.

In a cooperative research project techniques for the processing of digital elevation data have been developed. DEM's can be constructed, and are reported to be effective in automatic determination of relief features that can be useful in different methods of relief representation, and

in other branches of thematic application, e.g. land evaluation, micro-wave propagation, remote sensing, in mapping of geomorphology, erosion etc.

Due to the relatively rapid spread and development of the remote sensing centre of the Institute of Geodesy, Cartography and Remote Sensing, there is a constant stimulus for production and updating of land-use/land-cover maps. /There are no such standard maps in Hungary./

Based on LANDSAT Multi-Spectral Scanner and Thematic Mapper technology is being developed for natural resource assessment and the monitoring of land-use changes in agricultural areas first /appr. 70% of the country/. Classification methods have been tested for the verification of attribute data of the 200 meter resolution national digital terrain model of the country. The potential of the high resolution digital SPOT imagery, as opposed to traditional metric aerial photography, is presently under thorough investigation. Although Hungary is well covered with 1: 10,000 and even larger scale base maps, for historical reasons there are several data bases without a common geometric reference. The introduction of the Unified National Mapping System, however, facilitated the development of compatible, geocoded digital cartographic data bases.

There are thorough developments in some topographic data bases. First of all I mention the National Topographic Data Base for large scale of 1: 1,000 to 1: 4,000. This system has been used to complete two test areas for large towns, e.g. Szeged and Budapest to assist in the establishment of national digital cartographic data standards.

In addition to the previously mentioned data base, the development of the National Elevation Data Base at a scale of 1: 10,000 is being carried out with digitizing of horizontals.

Additionally, the Institute of Geodesy, Cartography and Remote Sensing led the design and implementation of the topographic data base containing 60,000 geographic names for all 19 Hungarian counties. The Institute of Geodesy, Cartography and Remote Sensing, along with Cartographia Enterprise, also took part in the development of the Cartographic Thesaurus; a non-spatial data base of approximately 30,000 thematic maps, an aid for compilation and production purposes. Numerous research efforts can be recognized as several users satisfy their specific needs from agriculture, regional planning, and industry. Nevertheless, most of these systems are not true GIS's, but serve as information systems for decision making with cartographic tools included.

The appearance of more distributed, PC-based systems improves the opportunity for cartographers to develop GIS technology for both local and national levels, e.g. at the Scientific Institute for Regional Planning, and the Research Institute for Soil Science. The Hungarian Soil Information System applies the geo-reference system of the Unified National Mapping System for all point, line and

and polygon data, with the development of multi-colour /i.e. non-binary/ quadtree techniques in a GIS environment, with input, data base management, query, and output subsystems. The system is being developed for Pest County.

Reliable and efficient information collection on the major crops in Hungary is of vital importance both at the decision making level of the Ministry of Agriculture and Food and at the level of farms too. Satellite remote sensing seems to be adequate to this task. A satellite based geographic information system for agriculture has been worked out in Hajdú-Bihar County. A complex processing system, REFER /System for Regional Inventory and Modelling of Natural Resources/ is introduced together with the role of its subsystems in data processing. This system is based on raster and vector subsystems. GIS's and semi-GIS's developed in the last couple of years in Hungary may be classified by the following groups: information systems for geodetic and cartographic purposes, town information systems, information systems for regional planning, information systems for different thematic and statistics analysis, information systems for CAD/CAM aims, decision making information systems mainly in Hungary in the interest of environmental protection.

In addition to the previously mentioned research and development that involves several aspects of data manipulation /e.g. remote sensing and DTM's/, some "Traditional" research concerning generalization using filtering and classification rules for choropleth maps are carried out at the Department of Cartography, Eötvös Loránd University of Sciences. Basic research was conducted for the application of spatial statistics in mapping natural phenomena.

Trends in Czechoslovakia

The development of computer cartography takes place within the framework of the state governed Department of Geodesy and Cartography, at the Universities J.E. Purkyne in Brno and I.A. Comenius in Bratislava, the Technical Universities in Prague and Bratislava, as well as at the Institute of Geography, Czechoslovak Academy of Sciences in Brno, and within the framework of numerous design offices. Verified technologies of cartographic production are centered primarily on the field of large scale cartography and on medium and small scale thematic map production.

The Department of Geodesy and Cartography develops the information system of geodesy and cartography with three fundamental subsystems. These are: -SIG, information about geodetic points; -REN, real estate information; -SLI, localization information.

The third one is the most important as it serves the automated production of state mapping work for basic large scale maps /1: 1,000, 1: 2,000 and 1: 5,000/. The -SLI subsystem represents essentially a digital map of selected localities stored on large capacity disks.

Thematic map production is coordinated by the Institute of Geography of the Czechoslovak Academy of Sciences, which is equipped with the DIGIKART graphic system of Czechoslovak production. Its components are: Digipos for semi-automatic digitizing, ADT computer and Digigraph. The most significant example of the production of thematic maps is the Czechoslovak Population Atlas and Atlas of the 1980 census of the Czech Socialist Republic, both developed by means of automated technologies. Automated technologies for thematic maps are also developed by regional institutes and universities. The system of data acquisition, encoding and storage is governed by policy and by the automation techniques available /system includes for instance GRADIS/.

A data base of a basic large scale map of the CSSR was developed; its fundamental entity is an object of the type "point, line and area". Every object is described by a classifier, a coordinate list and the drafting specifications. Attributes include information about the connections of particular points, the type of line, its thickness and the type of the point symbol according to the symbol key.

A number of computer graphics systems of both foreign and home provenance are used. Among the foreign systems the products of the firms CONTRAVES and KONGSBERG are predominant. Czechoslovak production is represented by the digitizers DIGIPOS and DGZ, the plotting tablets DIGIGRAF, AGS 4500, with the control computer ADT and the ISAP system with the SMEP computer are produced serially. The program language is mostly Fortran and PL/1, the graphics software is SFC, GFS and GRAFOS. The predominating systems are DOS-3 and DOS-4.

Trends in Poland

Geographic Information Systems are one of the most rapidly developing areas of geosciences in Poland. The progress in this field can be described with three kinds of activity. The first group includes the creation of universal tools for constructing GIS. Some theoretical and conceptual work belong to this group. The second direction is the elaboration of subject-oriented GIS for specific professional branches. The third one is based on application of ready commercial GIS for specific purposes.

In the Geodetic and Cartographic Data Processing Institute /now in the Institute of Geodesy and Cartography/ a geographical information system of universal purpose, based on regular grid, is being elaborated. They deal with data models, data base structures, modular spatial analyses, cartographic display and methods of data acquisition as well as of implementation based on elaboration of algorithms and programmes.

The representatives of the second direction of GIS development are the systems BIGLEB, elaborated by the Polish Soil Sciences Society and PROMEL, developed by the Irrigation Design Bureau. Both systems allow the collection of various data about soils, hydrology, elevation, crops etc. in a regular grid based on geographic coordinates. Captured data

are analyzed and processed for the purpose of spatial planning and designing.

The same direction of activities includes the project for the Land Information System /LIS/ of Poland elaborated at the Geodetic and Cartographic Data Processing Institute. It is based on multipurpose cadastre and digital base maps. The LIS will have many connections with branch information systems and will serve mainly administrative authorities on several levels.

One of the important tasks of GIS data base is the topographic information. In 1987, the concept of digital topographic map was prepared at the Geodetic and Cartographic Data Processing Institute. The concept took into account computer-assisted topographic mapping processes as well as supplying any GIS with topographic information.

Examples for the third kind of GIS projects may include the ERDAS system application for remotely sensed data processing at the Institute of Geodesy and Cartography. The implementation of ARC INFO system for environment protection projects is also planned.

The computer market in Poland is full of personal computers, and this is why they plan to create one option of geographical information systems for that kind of equipment. 32-bit personal computers with high capacity disks are sufficient for a series of applications, but systems to cover the territory of the country demands larger computers. For research work they use MicroVAX II computer, which could be exploited for one of the operational geographical information systems. The above mentioned systems, BIGLEB and PROMEL were implemented on Polish made ODRA minicomputers. There is no hardware oriented geographical information system like INTERGRAPH installed in Poland. The equipment they use consists of foreign and Polish digitizers like CODIMAT of Contraves /Switzerland/, or KARTOMETR of PCO /Poland/; plotters of Contraves /COROGRAPH DC2/, Hewlett-Packard /Draft Pro, HP 7585/, Roland /990/, IBM-compatible personal computers and MicroVAX II with 9 Mbyte memory and 491 Mbyte hard disk. For GIS project purposes they combine several of the mentioned devices to establish one compatible configuration.

Trends in the USSR

In the USSR we have witnessed an abrupt increase in the computer-assisted cartography and GIS; I will give you now some examples of some research centers at least.

A factual orientation of the research will be documented on the example of the Institute of Geography, Academy of Sciences, which is the centre for developing automated cartography and geoprocessing: elaboration of the conceptual framework of methodology and program software of geographical-cartographical modelling with utilizing numerical and logical and digital methods and geo-informational techniques, and at last but not least, elaboration of the theory and methods of global atlas mapping, including the digi-

tal expression of cartographic information and the experiment with forming banks of global cartographic data.

Up to the year 2000, the elaboration of methods of computer cartographic simulation is planned, the experiments with utilization of the GIS in modelling mutual relations and changes in territorial structures of the geosystems, cartographical data bases of local, regional and global levels.

From other centres where GIS of automated cartographical systems are developed, let me mention the Geographic Faculty of Moscow State University, where different advanced systems are oriented to the needs of thematic mapping. The technical framework consists of computers VAX, MITRA, the graphical display Radians 320, the digitizer and digigraph Benson, further they have a colour flat bed plotter. The supplied mathematical system is Grafiksi, specially created at the workplace for the needs of thematic mapping. A parcel of programmes is intended for preparing data including digitizing of cartographical materials, for revealing errors, for managing /their storage and removal for the processing/, logical-mathematical-statistical analysis, spatial approximations of regular and irregular networks, cartometric operations, constructing the system of cartographic signature. The aim of this system is to complete various maps for Moscow region, for schools and universities.

At the Faculty of Geography of Kazan' GIS is being developed for environment protection. It is intended for the observation, analysis and synthesis of complex geographical objects with acquiring forecasts. The system works on Soviet made hardware.

Very active in the development of GIS are the Baltic republics; it is right to speak about Estonian school of geo-information.

At "Kartografija" a project is being developed for an automated system of small scale map production. It is an information-technological man-machine system including all software and hardware elements for input, edit and output of 1:2,5 million scale map sheets covering the whole USSR. To mention the last example of the many systems developed in the USSR is the the GIS elaborated since 1981 at the Pacific Institute of Geography, Academy of Sciences, in Vladivostok. In 1986 the principles of compilation of the GIS for utilization of the environment were elaborated. In addition to numerous practical examples of its utilization, the following results were obtained: a computer atlas of temporal-spatial variability of agricultural production in the south of Far East has been compiled and issued, as well a series of computer-assisted maps of the dynamics for different thematic purposes. The system is being further developed taking into account the utilization of remote sensing and the methodology of processing dynamic images.

There were only few examples from the activity of Soviet computer-assisted cartography and GIS's.

COMECON COMMITTEE ON COMPUTER-ASSISTED CARTOGRAPHY

The work of the geodetic services of socialist countries is supported by the activity of special committees on cooperation in science and technology. These committees have been active for several years in increasing the effectiveness of scientific-technological development in the field of geodesy and cartography. The major topics include geodetic networks, remote sensing, complete automation of map-making, engineering geodesy etc.

The development of partial processes in the automation of large- and small-scale maps making was the primary importance in the field of map production automation.

New results in large-scale map production were mainly achieved in the development of automated data collection /both on the field measures and photogrammetric methods/ and interactive digitizing methods, and in the standardization of the requirements systems of large-scale maps automation.

Progress has been achieved in the automation of small-scale maps production: the standardization of the content of the cartographic thesaurus, and the automated preparation of the 1: 2,5 million world map.

In the forthcoming period larger effort will be laid on the automation of medium-scale or topographic maps. In this field the main purpose is the development of all hardware and software related to complex map production. The hardware components include the Czechoslovak digitizer, the computer and its peripheries made by the GDR, interactive work station, the Hungarian laser plotter etc. The system is of module configuration and it ensures both raster and vector processing of information, and it is also in accordance with the map standards of the COMECON.

The advantages of cooperation will only be remarkable in the long run.

SUMMARY

After the short description of the achievements in cartographic automation and geographic information systems in Hungary and in some of the Eastern European countries, the main trends are summarized as follows:

1. An increased demand for cartographic automation and GIS is recognizable in the countries of Eastern Europe.
2. Research, development and implementation are primarily carried out by their central authorities of cartography and geodesy, the enterprises and research institutes, but an important role can be attributed to the research institutes of academies and various ministries, and the technical development programmes of ministries.
3. The main directions of development are determined by the achievements of advanced industrialized countries known

through professional publications and conferences.

4. The automation of large-scale maps production is almost completed in most of the countries; and the same applies to some types of small-scale maps; the general situation is that only some of the partial processes are now automated, like in the case of medium-scale or topographic maps production; the common effort to automatize the preparation and revision of this category of maps has only just started on a conceptual level with COMECON hardware and software.

5. The countries concentrate large state resources and efforts on the development of various cartographic information systems /data bases/, but they are still rather in the state of development, experiment and research, and are not uniform and complete systems for the whole country.

6. As for the spatial or geographical information systems, they are produced in large number, they correspond to the centralized structure, but their vertical construction is usually incomplete. General purpose GIS's are becoming popular /e.g. ARC INFO, ERDAS/.

7. Western products dominate in hardwares /except computers/, though the countries have also started to produce hardware for cartographic purpose /Poland manufactures digitizers, the GDR produces interactive devices, Hungary has laser raster plotter etc./. Unfortunately, the level of raster input and output instruments production is rather lower at present. Their purchase is hindered by embargo.

8. The software development is largely influenced by the availability of abundant labour force /a cheap manware/; this is manifested in several and various kinds of GIS development projects. It often happens that the products are not compatible, which is explained by the fact that the commercial market in these countries is in fact missing. The same applies to the production of hardwares as well. In Hungarian-Polish relation various PC's are widely used; their software development for cartographic purpose is also advanced.

ERROR IN CATEGORICAL MAPS: TESTING VERSUS SIMULATION

Nicholas R. Chrisman
Department of Geography DP 10
University of Washington, Seattle WA 98195 USA
CHRISMAN@MAX.bitnet or CHRISMAN@MAX.ACS.WASHINGTON.EDU

ABSTRACT

Understanding error in maps requires a combination of theory (new models) and practice (understanding how error can be measured in real applications). While other research emphasizes mathematical models to simulate error, a practical test provides a more useful judge of cartographic data quality. A comprehensive test, overlaying two categorical maps intended to be the same, can provide an estimate of separate components of error including positional and attribute accuracy along with scale effects.

MAP ERROR: A FOCUS OF EFFORT

A few years ago, cartographic data quality and map error could be called a neglected topic (e.g. Chrisman, 1983). Recent developments have placed substantial attention on data quality, but most activity has focused on recognition that there is a problem. A number of components are required for overall improvement in the treatment of cartographic data quality. At the operational level, practitioners need tools to reduce error but tools require diagnostic tests. The tests, in turn, will reflect some model of error. Some such models can be imported from other sciences, but certain forms of cartographic information will require new models. This paper outlines a procedure to test one common form of cartographic data. The result is not a full-fledged "model" of error; it does provide a taxonomy of error which can lead to a model of error.

Fundamental differences over error

A major impediment to progress has been confusion over the understanding of cartographic error. The profession seems split into a number of incompatible schools of thought. A full treatment of intellectual history would have to begin with a review of the many disciplines which combine to contribute to modern GIS developments, but such depth would occupy the full length of this paper. Instead, this paper will concentrate on providing an alternative to one dominant approach to cartographic error models.

The Simulation School. One group of researchers (exemplified by Goodchild and Dubuc 1987, but including others as well) seeks to develop a procedure which can produce a "random" map. Their approach adopts common stochastic modeling methods from mathematical statistics. Such modeling can construct some numerical procedures with results that share certain measures (topology, size distribution)

with actual maps. The goal of this research seems to be developed by analogue from other sciences where a generalized random model could be developed to create analytical tools for a broad class of information. Perhaps this research track will lead to a generalized model of error, but such success is bound to be far off. Constructing a simulation that produces plausible maps does not mean that real maps arise from that process or share similar mathematical properties.

Other sciences engage in stochastic modeling from a firm foundation in measuring their phenomena of interest. In particular, the definition of error in biomedical or agricultural experiments is not a matter of controversy. The bulk of mathematical statistics depends on the concept of a "population"- a large or infinite pool of individual cases that will behave in essentially identical fashion. Error models can predict the probability of obtaining certain results from samples drawn from the population. For many sciences, the case and population paradigm summarizes potential error. By contrast, the mapping sciences have not developed a comprehensive taxonomy of what errors occur and what processes control the amount of particular kinds of error. The use of cartographic information in geographical analysis involves many properties (particularly colocation and other geometric properties) not considered in standard statistical treatment. In my opinion, it is premature to develop stochastic models for a field without a clear understanding of the fundamentals.

An Alternate Philosophy. Philosophically, stochastic modeling fits into an idealist view where the pure nature of things is clouded by an imperfect world. In other terms, the abstract model is more perfect and correct than the phenomenon it represents. While this view of the world has been held by some prominent philosophers for millenia, it is not the only possible approach. My philosophic position can be summarized by a few principles. I do not presuppose some ideal world which is more pure and correct. Observation, measurement, experiment and experience provide access to an inscrutable world. As humans we develop concepts, theories, and languages to organize our knowledge, but these human constructions are mainly useful in making further predictions of the actual operation of the world. Abstract, self-contained systems like mathematics or programming languages can be absorbing, but they prove their utility by allowing humans to manipulate the real world. While this philosophy (perhaps formally termed pragmatic realism) may sound non-controversial or banal, it leads to a different approach to cartographic error. I believe that the ultimate arbiter of cartographic error is the real world, not a mathematical formulation. I define error as the deviation of our representation from the actual state of affairs. This deviation will vary from place to place and from time to time and from technology to technology. I can only use a mathematical model to predict this error when I organize the evidence to generalize from the specific case.

Just as there are inductive and deductive approaches to scientific method, there are distinct approaches to cartographic error. In some earlier papers (Chrisman, 1982b), I adopted a rather deductive strategy of assigning error to each step performed from source material to final digital product. Eventually this approach must be adopted for routine estimation of data quality, but it is not the appropriate strategy for

developing a taxonomy of error behavior. This paper is based on the use of testing and empirical evidence to help structure a theory of error.

Geometry and Attributes

Perhaps the most commonplace distinction in cartography and GIS contrasts treatment of strictly spatial data from the rest of the aspatial context. The spatial elements are best termed "geometry", a term which includes both metrical position and topological components, though other terms are in common use. There is also diversity in terminology for the aspatial "thematic" components. This paper will use the term "attribute", although it often covers both geometric and thematic components.

It is a common trap in cartography and other sciences to seek finer and finer nuances of terminology as a substitute for theoretical insight. Whether geometry is simply another attribute or must be treated differently is one of the major issues dividing current GIS implementations. To date the debate between a dual "georelational" approach (Morehouse, 1985) and a unitary representation (e.g. Charwood and others, 1988) has focused on efficient use of computers. Although such efficiency has been a primary measure for GIS, these considerations have not included treatment of data quality. Some research on error models follows the traditional division of attribute from geometry while others seek unitary models. In the simulation school, the fundamental tools deal with continuous surfaces which place the thematic component in a common metric with the horizontal position. With a surface model, there are many mathematical operations which are quite valid, but the behavior of surfaces is not the only problem confronted by cartographers. This paper will seek to show that a dual approach is required for some forms of map data.

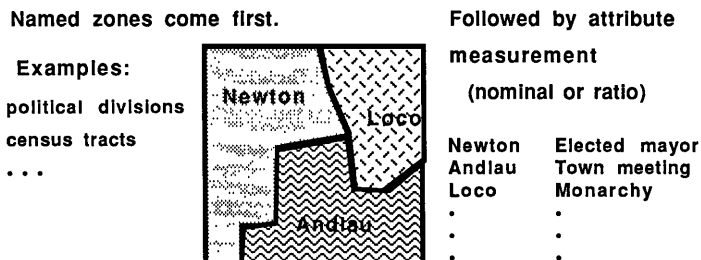
TYPES OF CATEGORICAL DATA

Eventually, there is a need for a model of error treating all forms of spatial information. Much of the work in the mapping sciences has treated the positional accuracy of "well-defined points". Such objects can be treated separately without worrying about their context. The current interest in cartographic "feature" data adheres to this simple world where objects are surrounded by the void. It may be possible to construct an error model for feature data from more traditional mathematical statistics because features do not involve topological properties and other two-dimensional characteristics. However, I believe that feature data is often selected from a richer view of spatial relationships.

An important property of spatial information is exhaustiveness. Most analytical cartography has focused on surfaces, exhaustive fields of continuous varying attributes, but this form of data, while mathematically tractable, does not cover all of the problems faced in GIS application. The most complex problems arise when the thematic information - the attributes - are measured on a categorical scale of measurement, either nominal or ordinal. This paper is primarily concerned with one kind of two-dimensional distribution, termed a *categorical coverage* (Chrisman, 1982a). A categorical coverage is a specific type of polygon map used quite frequently for GIS applications.

It is important to distinguish this form of polygon data from spatial collection units (Figure 1).

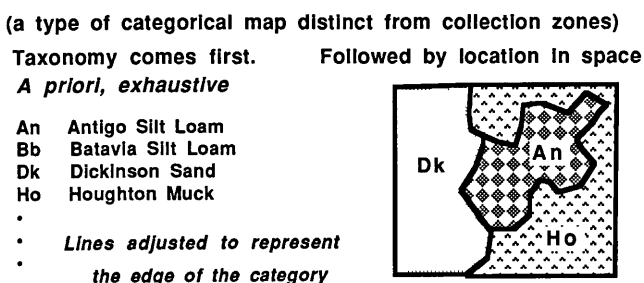
Figure 1
Arbitrary collection units
 (one type of categorical map)



The important consideration is which component, the spatial description or the attribute, takes logical precedence (Sinton, 1978). In the pure case (administrative units such as municipalities), the positional description of the object precedes any attributes assigned. These maps are *choropleth* maps in the purest sense, because the places exist, then they are filled. [Choropleth has now come to refer to categorical maps derived from classed continuous distributions, but that does not alter the etymology.]

Many of the users of GIS software do not rely upon collection unit sources. The layers fed into a GIS are more likely to be soil maps, vegetation maps, ownership parcels, and many more. Although the distinction is not absolute, these maps derive from a different approach (Figure 2).

Figure 2
Categorical Coverages



Both forms of data (Figures 1 & 2) may be displayed as choropleth maps, but similarity of graphic display obscures fundamental differences. In Figure 2, some system of classification (the soil taxonomy, the vegetation classes, and even the list of taxable parcels) logically precedes the map. The map results from assigning each portion of the area into one class or another. Issues of positional accuracy, scale and other cartographic concerns become much more prominent than they are in the collection

zone case. The model of error implicit in collection units (spatial autocorrelation) relies on an underlying continuous distribution, aggregated into discrete and arbitrary spatial units. A model of error for categorical coverages reverses the logic. Spatial units are adjusted on a continuous space to reflect the categorical distinctions.

A FRAMEWORK FOR MEASURING ERROR

Before a complete stochastic model can be developed, the first step is to define the error to be modelled. The various disciplines involved in mapping have used widely varying concepts of error, and each should make a contribution to a comprehensive model. The fundamental issue in statistics is understanding deviations. The deviations possible in a categorical coverage involve diverse components. In particular, there are positional (geometric) issues and attribute issues. The concept of deviation used for these two are usually quite different, but, in a categorical coverage, the various error components interact. Goodchild and Dubuc (1987) reject the separation of geometry and attributes, but there are strong suggestions that parallel treatment is useful. This section describes a mechanism to deconvolve spatial error into identifiable processes, each with distinct mathematical treatment.

It is relatively easy to catalogue all of the steps used to create spatial information. Each of these steps no doubt introduces different types and amounts of error in the resultant products. But a complex model of this kind (essentially the proposal of Chrisman, 1982a) is quite difficult to verify. The amount of error can be best ascertained by a process of testing. Tests have inherent limits in their ability to distinguish errors from different sources. The existing practice of mapping sciences include a very few established testing procedures. Taken together, these tests do provide some sort of coverage for the range of problems included in the proposed US National Standard. Some of the most recent tests, like the tests of topological integrity (White, 1980), have developed from the introduction of computing to mapping, but most are longer established. The positioning sciences (geodesy, surveying, photogrammetry) have tests of positional accuracy based either on repeated measurements (internal evidence) or on tests against an independent source of higher accuracy. In both cases the tests treat "well-defined points", cartographic features taken in isolation. Photointerpretation and remote sensing use point sampling to test classification accuracy, following some relatively standard procedures to estimate proportions of a categorical variable (Rosenfield and Melley, 1980). The spatial sampling techniques outlined by Berry and Baker (1968) still provide the spatial logic for these tests.

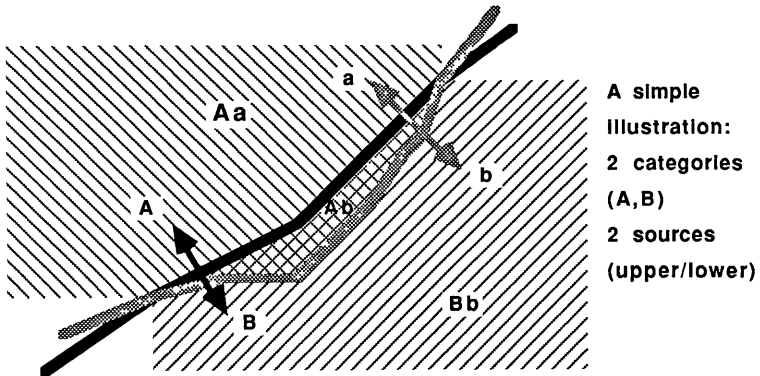
The current range of tests are each designed to treat a distinct component of overall quality. Thus they can easily be fooled by error of the other components. For instance, the emphasis on well-defined points in positional tests is to reduce the impact of classification error. While it may be correct to isolate some components for some purposes, there is a need for a comprehensive test, particularly for exhaustive categorical coverages. The common point sampling approach to classification accuracy can fail to distinguish between errors in positional and attribute components.

A comprehensive test compares complete maps, not just sampled locations. Two categorical coverages purporting to map the same phenomenon are overlaid comprehensively, and the results form a test of accuracy. This test has been applied in some isolated circumstances (for instance, Ventura and others, 1986), and it has been accepted as an alternative to point sampling in the proposed US National Standard.

If one source is assumed "correct", it is a test of the other, but it could also be a test of repeatability. As in many statistical applications, a test pairs every point on the map by location on the ground. Such an arrangement, with an infinite number of points, requires a different error model than a "case" oriented approach. This framework is described incrementally, starting from some simple cases, then providing more complexity.

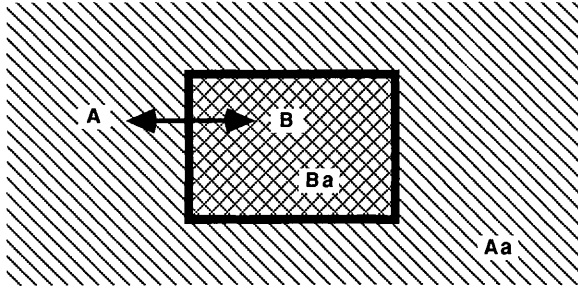
The most common form of error in overlaid maps is called a "sliver". As demonstrated in Figure 3, a simple sliver occurs when a boundary between two categories is represented slightly differently in the two source maps for the overlay. A small, unintended zone is created. Goodchild (1978) reports that some systems become clogged with the spurious entities that provide evidence of autocorrelation at different levels. These reports are a part of the unwritten lore of GIS, because most agencies are unlikely to report on failures. Some algorithms for overlay include a filter to remove the smallest of these, up to the level a user is willing to tolerate (Dougenik, 1980).

Figure 3
Slivers: the classic form of overlay error



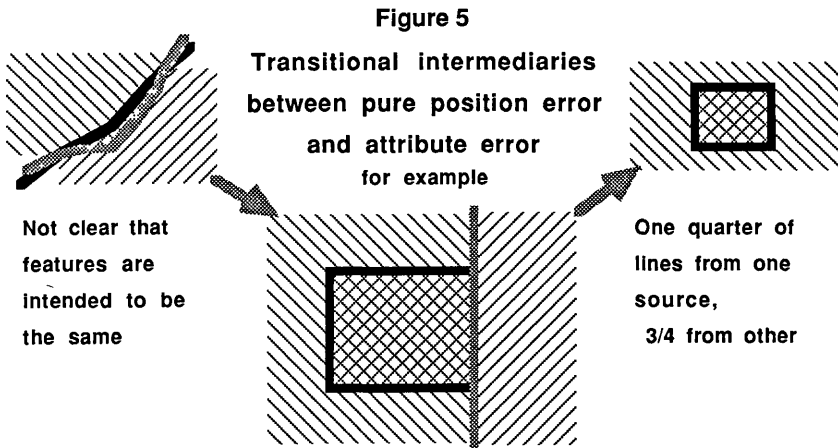
Although sliver error is the most frequently mentioned, an overlay test can discover other forms of error. To follow the example described above (comparison of two maps assumed to be the same), it would be possible to have a feature on one map source which is completely missing on the other, as shown in Figure 4. While the sliver error seems to arise from positional error, a missing polygon is caused by classification error. Unlike the rudimentary "feature" approach, a misclassification in a coverage assigns the area to some other category. Taxonomic similarity of the two categories could be modelled in some continuous phase space - as proposed by Goodchild and Dubuc (1987), or otherwise.

Figure 4
Another case of overlay error



As extremes, the positional sliver and the attribute classification error seem perfectly distinct. But the two are quite difficult to disentangle in practice. For instance, a sliver error might arise from an interaction of positional error and difficulty in discriminating the classifications (more of an attribute problem than an error in positioning technology). Chrisman (1982a) proposes a division of classification error for categorical coverages into components of *discrimination* (essentially the sliver effects) and *identification* (essentially the subject of Figure 4). This test seeks to build this distinction into a larger framework.

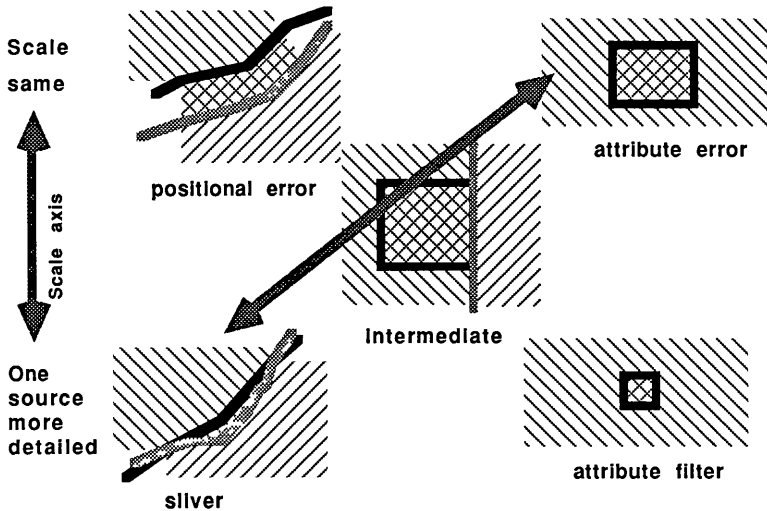
As an additional complication, not all error falls perfectly into the two cases presented in Figures 3 and 4. The sliver involves roughly the same contribution of linework from each source, while the classification error has all the linework from one source. As Figure 5 shows, there is a continuum possible between the two extremes which might be hard to classify. While it is easy to develop anecdotes about this kind of error, there is no workable theory in common use.



The previous argument deals with the existence of both positional and attribute error, but ignored the issue of scale. In spite of the power of modern GIS software, the basic information is still strongly dependent on scale. Positional accuracy of lines is expected to be linked to scale, but

the amount is rarely specified or measured. Even more so, attribute error is linked to scale. At some scales, features like farmsteads are consciously removed from land use maps. Scale involves a distortion of the information, but a distortion that is tolerated and expected. To develop a framework, Figure 6 shows how positional and attribute error might interact schematically with scale issues.

Figure 6
Scale adds a dimension to the transition
(previous diagram now the diagonal)



The framework presented above is more than a diagram. It provides the basis to construct a mathematical model where the total error is decomposed into a set of stochastic processes operating simultaneously. The stochastic process for boundary error will have to reflect the geometric impact of cartographic representation and processing, while the attribute error will have to reflect taxonomic similarity of classes. For the technical "process" errors that simply degrade the positional accuracy, the epsilon model (Perkal, 1966; Chrisman, 1982a; 1982b) may provide a useful start. For errors in identification or misclassification, some modification of Goodchild's phase spaces may be developed, depending on the basic science for the particular information. To further complicate affairs, these two processes will operate inside scale-dependent rules that can be modelled as filters and other constraints. It is extremely unlikely that we can expect a single overarching scheme to treat error in geographic information, but the constraints of testing must influence our ability to discern and differentiate such errors.

The framework developed above may explain the results obtained from empirical accuracy experiments. The concept of an exhaustive test through polygon overlay has been accepted as a component of the US proposed national standard, but few tests have been performed using this approach (for example, Ventura and others, 1986). Empirical results measure the total error from all processes, and there is no guaranteed mechanism to deconvolve them. Each of the individual

components above will be easier to model in isolation, then the error components can be combined.

CONCLUSIONS

Considering the public and private investment in geographic information systems, additional research on the error of overlaid maps is required. This paper sketches a preliminary taxonomy of error that can be used as the basis for research. With substantial development, a new set of analytical procedures may be developed, perhaps even a "geographical analysis of variance" (Warntz, 1966).

ACKNOWLEDGEMENTS

The research in this paper was supported in part by National Science Foundation Grant SES 87-22084.

REFERENCES

- Berry, B.J.L. and Baker, A.M.** 1968: Geographic sampling. In Berry and Marble (ed.) *Spatial Analysis*, 91-100.
- Charlwood, G., Moon, G. and Tulip, J.** 1987: Developing a DBMS for geographic information: a review, *Proc. AUTO-CARTO 8*, 302-315.
- Chrisman, N.R.** 1982a: Methods of spatial analysis based on error in categorical maps. PhD thesis, U. of Bristol.
- Chrisman, N.R.** 1982b: A theory of cartographic error and its measurement in digital data bases. *Proceedings AUTO-CARTO 5*, 159-168.
- Chrisman, N.R.** 1982c: Beyond accuracy assessment: correction of misclassification. *Proc. International Society of Photogrammetry and Remote Sensing Commission IV, 24-IV*, 123-132.
- Chrisman, N.R.** 1983: The role of quality information in the long-term functioning of a GIS. *Proceedings AUTO-CARTO 6, 2*, 303-321.
- Chrisman, N., Mezera, D., Moyer, D., Niemann, B., Vonderohe, A.** 1984: Modernization of routine land records in Dane County, Wisconsin: implications to rural landscape assessment and planning, *URISA Professional Paper 84-1*.
- Dougenik, J.A.** 1980: WHIRLPOOL: a processor for polygon coverage data. *Proc. AUTO-CARTO IV*, 304-311.
- Goodchild, M.** 1978: Statistical aspects of the polygon overlay problem, in vol. 6 G. Dutton, ed., *Harvard Papers on Geographic Information Systems*, Addison Wesley.
- Goodchild, M. and Dubuc, O.** 1987: A model of error for choropleth maps with applications to geographic information systems. *Proc. AUTO-CARTO 8*, 165-174.
- Harding, E.F. and Kendall, D.G.** 1974: *Stochastic Geometry*, John Wiley.
- Morehouse, S.** 1985: Georelational data structure for GIS, *Proc. AUTO-CARTO 7*.
- Perkal, J.** 1966: On the length of empirical curves. *Discussion Paper 10*, Michigan InterUniversity Community of Mathematical Geographers.
- Peucker, T.K.** 1976: A theory of the cartographic line. *International Yearbook of Cartography*, 16, 134-143.
- Rosenfield, G. and Melley, M.** 1980: Applications of statistics to thematic mapping. *Photogrammetric Engineering and Remote Sensing*, 46, 1287-1294.
- Sinton, D.** 1978: The inherent structure of information as a constraint to analysis: mapped thematic data as a case study, in vol. 7 G. Dutton, ed., *Harvard Papers on Geographic Information Systems*, Addison Wesley.
- Ventura, S. Sullivan, J.G. and Chrisman, N.** 1986: Vectorization of Landsat TM land cover classification data. *Proceedings URISA, 1*, 129-140.
- Warntz, W.** 1968: On the nature of maps. *Discussion Paper 12*, Michigan InterUniversity Community of Mathematical Geographers.

MODELING ERRORS FOR REMOTELY SENSED DATA
INPUT TO GIS

Michael F. Goodchild
National Center for Geographic Information and Analysis
University of California
Santa Barbara, CA 93106

Wang Min-hua
Department of Geography
University of Western Ontario
London, Ontario N6A 5C2, Canada

ABSTRACT

Different views of spatial resolution and accuracy present a major obstacle to the integration of remote sensing and GIS. Accuracy in remote sensing is modeled using probabilities of class membership in each pixel; in vector-based GIS it is modeled using concepts such as the epsilon band. The problem of linking the two views of accuracy reduces to one of realizing a stochastic process which must satisfy conditions of prior and posterior probabilities, and spatial dependence. We propose two suitable methods, one storage intensive and the other computationally intensive. The methods can be adapted to incorporate various forms of prior knowledge.

INTRODUCTION

Remotely sensed imagery provides a fast and efficient means of collecting large volumes of information about the earth's surface. Raw spectral responses can be registered, corrected and interpreted using sophisticated image processing systems, and a variety of methods of pixel classification have been developed to transform imagery into rudimentary maps for such themes as land use or vegetation cover. The response recorded for each pixel in a particular band is an integral over the area of the pixel of a continuous, spatially autocorrelated variable, and it is common to think of response data as a random sampling of a continuous surface or field. On the other hand a classified image can be conceptualized as an array of discrete values in which each pixel has been assigned to one of a number of classes.

A GIS can be defined as a system for input, storage, analysis and output of spatial information. As such, its main strengths lie in its ability to give the user access to an apparently scale-free, seamless electronic map, to analyze simultaneously different layers or coverages of the same area, to measure the lengths and areas of geographical objects, and to allow easy updating and editing. The capabilities of a GIS can greatly extend the usefulness of a classified, remotely sensed image by allowing access to other data either to improve the accuracy of classification, or to enhance the range of possible analyses. On the other hand remote sensing has much to offer GIS as a source of

easily updated and low cost input data. For these reasons numerous attempts to integrate remote sensing and GIS have been described in the literature.

Vector-based GISs model the world as populated by objects, specifically classes of points, lines or areas. Land cover is often modeled in such a system as a class of non-overlapping areas which exhaust the study space, each area being associated with one or more attributes which describe its land cover class. In practice the use of this model is largely independent of the means by which the data was acquired, whether by digitizing the lines on an existing map of land cover, scanning an existing map and vectorizing the resulting image, or using an image processing system to classify and vectorize a remotely sensed image. However the appearance of the data may reveal the source, as pixel edges will likely still be evident in a layer obtained from remote sensing, unless the lines have been subsequently smoothed.

Although there are undoubtedly significant technical problems in interfacing remote sensing and GIS, we wish to argue in this paper that the conceptual problems of interfacing systems which view the world respectively as fields and objects are in the long run more challenging, and will be a more substantial obstacle to the use of remotely sensed images in object-based systems. Our purpose in this paper is to explore the implications of such interfacing from the perspective of the interrelated issues of spatial resolution, error and accuracy. More specifically, the paper examines the extent to which concepts of error in imagery can be related to corresponding concepts of error in objects. The paper expands on work previously described by Goodchild and Wang (1988).

The next section reviews recent efforts to deal with the problem of uncertainty in object-based GIS. This is followed by a discussion of error in classified imagery, and by a review of techniques which can be used to interrelate these two views of the accuracy of spatial information.

ERROR IN OBJECT-BASED GIS

The use of high precision digital processing on data of undetermined accuracy has inevitably raised awareness of the problems of error in spatial data handling in recent years (see for example Walsh, Lightfoot and Butler 1987; Burrough 1986), besides leading to specific artifacts such as sliver polygons (Goodchild, 1979) and conflicts between geometry and topology (Franklin, 1984). Problems are made particularly acute by the ease with which a GIS can be used to change the scale of data without a corresponding change in its spatial resolution, and by the degree to which GIS processing of data from multiple sources distances the user from the data collection and interpretation process. As a result the users of GIS products are often unaware of the uncertainties and caveats which surround any spatial information.

Statistical models of the uncertainty in the locations of

point objects are well developed in surveying and geodesy. However their extensions to more complex line and area objects are not straightforward for several reasons. A model of the relationship between a true line on the ground and its representation as a series of digitized points and connecting straight lines in a spatial database must include not only the correlations which exist between errors at neighboring points (Keefer, Smith and Gregoire 1988), but also the process by which the points themselves were selected by the digitizer operator.

Despite these difficulties, simple approaches to describing accuracy of object representations can be found in the concepts of tolerance and error bands used in many digitizing and overlay systems. The Perkal epsilon band (Perkal 1956; Blakemore 1984) is a buffer of width epsilon on either side of a line or polygon boundary. In digitizing, two lines can be assumed to join and are consequently 'snapped' together if one lies within the other's epsilon band; similarly, in overlay, a line on one map which lies within the epsilon band of a line on the other map is assumed to represent the same line on the ground, and any associated sliver polygons are therefore removed. Unfortunately this deterministic view of the epsilon band can produce unwanted results in the following way. Line A can be found to lie within line B's band, indicating that A and B are the same; A can lie in C's band indicating that A and C are the same; but C can lie outside B's band. In this situation it is easy to generate inconsistencies, particularly if the positions of objects are adjusted in operations such as snapping. A probabilistic version of the concept could potentially resolve such problems.

The process of digitizing tends to result in errors and distortions which are substantially constant over a map, and depend only on the scale at which the map was digitized. On the other hand other, often more significant sources of error are unfortunately not as constant. It is common to distinguish between processing errors, which include those introduced during digitizing, and the source errors which exist between the source document and the reality which the document models. In the case of a land cover map these include the inaccuracies which result from modeling a complex pattern of spatial variation with a relatively small number of homogeneous areas separated by sharp discontinuities; in reality areas are not homogeneous and boundaries mark zones of transition rather than sharp breaks. Although it may be possible to model many forms of processing error (see for example Amrhein and Griffith 1987; Keefer, Smith and Gregoire 1988), it is virtually impossible to describe source errors without access to additional information such as ground surveys.

ERROR IN CLASSIFIED PIXELS

Many methods of image classification estimate the probability that a pixel belongs to each of a set of possible classes: commonly, the class to which the pixel is

finally assigned is that having the largest probability. However the complete set of probabilities for each pixel constitutes a useful source of information on the uncertainty of classification. Let the subscript i denote one of the n pixels, j denote one of the m classes, and let the vector $\{p_{i1}, p_{i2}, \dots, p_{im}\}$ denote the set of probabilities for pixel i . Let $M_i, M_i = k \mid p_{ik} > p_{ij} \forall j \neq k$, be the most likely class.

A maximum likelihood classification based on M_i allows easy restructuring of the pixels to objects using a raster/vector conversion algorithm, but it implicitly deletes all potentially useful information on uncertainty, thus creating the kind of situation we have already described as common for object-based models of such themes as land cover. We propose instead that the entire vector be passed to the GIS, allowing GIS analysis to incorporate uncertainty into its processes and products. In most applications it is likely that only a small proportion of the m probabilities for each pixel will be significantly large, so we need not necessarily assume that this strategy will result in an m -fold increase in the storage requirements of this particular layer.

In order to obtain objects from the vectors of probabilities we must first create a realization, or a specific outcome of the stochastic process which the probabilities define. Let X_i denote the class to which pixel i is assigned in a particular realization: the maximum likelihood classification generates an outcome of the stochastic process by simply assigning $X_i = M_i$. The same set of probabilities can be used to produce multiple realizations or outcomes, corresponding to the tossing of a dice, and the differences between outcomes represent uncertainty.

The simplest realization would be a multinomial process in which the outcome in each pixel is determined independently, based on the known probabilities. A simple approach would be to generate a random number $x_i, 0 \leq x_i \leq 1$, and assign class k if:

$$\sum_{j=1}^{k-1} p_{ij} < x_i \leq \sum_{j=1}^k p_{ij} \quad (1)$$

However the result would appear unreasonably fragmented because of the independence of the outcome in neighboring pixels, and it is very unlikely that large, homogeneous patches of similarly classified pixels would develop except where one probability is close to 1 and classification is therefore almost certain. This process would fail therefore to model the common situation in remote sensing where a large patch of many pixels returns a homogeneous spectral response, but nevertheless has a very uncertain classification. A further objection is that by ensuring homogeneity within pixels but independence between them, we create a result which is very dependent on pixel size.

These objections can be removed if the outcomes in neighboring pixels are allowed to be correlated. In essence, we require a process of realization in which two

properties are satisfied: a) the proportion of realizations in which pixel i is assigned class j tends to p_{ij} as the number of realizations tends to infinity (posterior and prior probabilities are equal), and b) outcomes within one realization display a prescribed level of spatial dependence.

METHODS OF REALIZATION

Goodchild and Wang (1988) described a process in which each pixel was first independently assigned to a class. This initial image was then repeatedly convolved with a low-pass filter, in order to impose spatial dependence (see also the ICM technique of Besag 1986). The paper illustrated the use of a 3 by 3 filter with the rule that in each pass the central pixel was assigned the modal class of the 9 pixels within the filter window. This process was demonstrated to generate spatially dependent realizations, allowing uncertainty in pixel classifications to be converted to uncertainty in the location of objects and to concepts such as the epsilon band. However it is easy to show that the low-pass filter generates posterior probabilities which are not equal to the prior probabilities, violating the first requirement above, except in special cases.

Cross and Jain (1983) have described a process of modeling spatially dependent images in which an initial set of outcomes, such as that produced by our simple multinomial process above, is modified by selectively swapping the contents of randomly selected pairs of pixels (see also Goodchild 1980). Again, while the result is a pattern which has strong spatial dependence, in general the prior and posterior probabilities in each pixel are not equal.

Two approaches appear to offer a way of satisfying both requirements simultaneously, one computationally intensive and the other storage intensive. The latter is conceptually simpler and will be described first. Let q denote a number of realizations, say 100, and suppose that initial classes are assigned to each pixel in each of q realizations by the multinomial, spatially independent process. The proportion of realizations in which a given pixel is assigned to a given class will be approximately equal to the prior probability. Now suppose that some means exists to measure the level of spatial dependence present in any one realization, and that a target value for this measure has been defined. Suitable measures can be found in the literature on indices of spatial autocorrelation (Cliff and Ord 1981; Goodchild 1988). The technique then executes the following steps until the target is reached, or no further improvement can be obtained:

```
select a pixel at random;
for that pixel, select a pair of realizations at random;
if the pixels are currently assigned to different
classes, then;
    swap the contents of the pixels if by so doing the
    recomputed measure is closer to the target;
end if;
```


repeat.

Because the technique cannot change the numbers of each class assigned to any one pixel across the set of realizations, we ensure that the posterior and prior probabilities are equal.

The second method implements a spatially autoregressive process in which the value in any cell is correlated with the value in nearby cells (Haining, Griffith and Bennett 1983). A spatially autoregressive process on a lattice can be defined as:

$$\mathbf{z} = \rho \mathbf{W} \mathbf{z} + \epsilon \quad (2)$$

where: z_i is the value assigned to pixel i by the process;
 ρ is a spatial autocorrelation parameter;
 \mathbf{W} is an n by n array of interactions between pixels;
 ϵ_i is an independent, normally distributed error term with zero mean and variance σ^2 .

We assume that W_{ij} is 1 if pixels i and j are 4-adjacent, else 0. The solution for \mathbf{z} is given by:

$$\mathbf{z} = (\mathbf{I} - \rho \mathbf{W})^{-1} \epsilon \quad (3)$$

The \mathbf{z} are known to be multivariate normal with zero mean and with variance-covariance matrix given by (Haining, Griffith and Bennett 1983):

$$\sigma^2 [(\mathbf{I} - \rho \mathbf{W})^T (\mathbf{I} - \rho \mathbf{W})]^{-1} \quad (4)$$

We can obtain a class X_i for pixel i from the following rule:

$$X_i = k \text{ iff } F(z_i) < p_i \quad (5)$$

where $F(z)$ is the probability that an independent, normally distributed random number with mean 0 and variance given by the diagonal terms of (4) exceeds z .

Unfortunately the technique requires the inversion of an n by n matrix, and special methods are necessary to generate realizations in arrays of more than about 8 by 8 pixels.

Both techniques have the advantage that it is easy to include prior information about such objects as field boundaries, roads or water. The spatial dependence between pixels across a significant boundary can be deleted by setting the appropriate terms in \mathbf{W} to zero instead of 1, which will cause the boundary to emerge in each realization. In the swap technique the same effects can be achieved by setting appropriate terms in the evaluation function, which will in most cases include the equivalent of the \mathbf{W} matrix. Similarly the presence of known classes such as water can be dealt with by setting the associated probability to 1 and all others to 0 in affected pixels.

DISCUSSION

The techniques described can be used to simulate the effects of uncertainty in both field and object views of spatially varying phenomena. Goodchild and Wang (1988) illustrate the generation of a cross-classification matrix, which is the approach often used in image processing to assess accuracy, and the sliver polygons and epsilon bands of the object-based approach to accuracy. Although these methods emphasize the equivalence between the measures used in both views, we stress that it is the field-based probabilities which are externally generated, while the object-based measures must be derived from them. This serves to emphasize the earlier point that an object-based view of spatial data rarely carries information on which an objective model of accuracy can be based.

The techniques provide a framework within which it is possible to discuss a number of conceptual models of uncertainty in spatial data. We have argued that the independent pixel is almost always inappropriate: because of spatial dependence these techniques produce patches whose size and shape are controlled by user-defined parameters and largely independent of pixel size. By setting appropriate levels of spatial dependence it is possible to produce a range of outcomes from highly fragmented and scattered patches when spatial dependence is low and local, through large patches which result from the aggregation of numbers of spatially dependent choices. With high levels of spatial dependence and with appropriately set terms in the W matrix, it is possible to have predefined patches in which the outcome is essentially the result of a single trial, thus simulating the example of the multi-pixel field whose entire class is uncertain.

We have thus far assumed that spatial dependence is a stationary property of the entire array. In reality some classes display patches which are more fragmented than others, and spatial dependence also varies from one region to another. In the future we hope to develop methods which will successfully simulate these conditions as well.

REFERENCES

- Amrhein, C. and D.A. Griffith, 1987, GIS, Spatial Statistics and Statistical Quality Control: Proceedings, IGIS '87, ASPRS/ACSM, Falls Church, VA.
- Besag, J., 1986, On the Statistical Analysis of Dirty Pictures: Journal of the Royal Statistical Society B48:259-302.
- Blakemore, M., 1984, Generalization and Error in Spatial Databases: Cartographica 21:131-9.
- Burrough, P.A., 1986, Principles of Geographic Information Systems for Land Resources Assessment, Oxford.
- Cliff, A.D. and J.K. Ord, 1981, Spatial Processes: Models and Applications, Pion, London.

Cross, C.R. and A.K. Jain, 1983, Markov Random Field Texture Models: IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-5:25-39.

Franklin, W.R., 1984, Cartographic Errors Symptomatic of Underlying Algebra Problems: Proceedings, International Symposium on Spatial Data Handling, Zurich, University of Zurich, 190-208.

Goodchild, M.F., 1979, Effects of Generalization in Geographical Data Encoding: in H. Freeman and G.G. Pieroni, editors, Map Data Processing, Academic Press, New York, 191-206.

Goodchild, M.F., 1980, Simulation of Autocorrelation for Aggregate Data: Environment and Planning A 12:1073-81.

Goodchild, M.F. and M.-H. Wang, 1988, Modeling Error in Raster-Based Spatial Data: Proceedings, Third International Symposium on Spatial Data Handling, Sydney, IGU Commission on Geographical Data Sensing and Processing, Columbus, Ohio.

Goodchild, M.F., 1988, Spatial Autocorrelation, Concepts and Techniques in Modern Geography No. 48, GeoBooks, Norwich.

Haining, R.P., D.A. Griffith and R.J. Bennett, 1983, Simulating Two-Dimensional Autocorrelated Surfaces: Geographical Analysis 15:247-53.

Keefer, B.J., J.L. Smith and T.G. Gregoire, 1988, Simulating Manual Digitizing Error with Statistical Models: Proceedings, GIS/LIS '88, ASPRS/ACSM, Falls Church, VA, 475-83.

Prekal, J., 1956, On Epsilon Length: Bulletin de l'Academie Polonaise des Sciences 4:399-403.

Walsh, S.J., D.R. Lightfoot and D.R. Butler, 1987, Recognition and Assessment of Error in Geographic Information Systems: Photogrammetric Engineering and Remote Sensing 53:1423-30.

CONCEPTS OF SPACE AND SPATIAL LANGUAGE

David M. Mark and Andrew U. Frank
National Center for Geographic Information and Analysis

Department of Geography, SUNY at Buffalo
Buffalo NY 14260 (Mark)
Department of Surveying Engineering, University of Maine
Orono ME 04469 (Frank)

BIOGRAPHICAL SKETCHES

David M. Mark is a Professor of Geography at SUNY at Buffalo, and holds a Ph.D. in Geography from Simon Fraser University (1977). He is immediate past Chair of the GIS Specialty group of the Association of American Geographers, and is on the editorial boards of *The American Cartographer* and *Geographical Analysis*. He also is a member of the NCGIA Scientific Policy Committee. Mark's current research interests include GIS, cartography, cognitive science, way-finding, and expert systems. Electronic mail: geodmm@ubvmc.bitnet.

Andrew U. Frank is New England Section ACSM Professor in Land Information and Associate Professor of Surveying Engineering at the University of Maine. He is an Associate Director of the NCGIA. Frank holds a doctoral degree from ETH-Zurich. His research interests include geographic information systems, object-oriented programming, and formal systems for spatial relations and objects. Electronic mail: frank@mecan1.bitnet.

ABSTRACT

Development of a comprehensive model of spatial relations is important to improved geographic information and analysis systems, and also to cognitive science and behavioral geography. This paper first reviews concepts of space. A critical distinction is between small-scale spaces, whose geometry can be directly perceived through vision and other senses, and large-scale space, which can be perceived only in relatively small parts. Fundamental terms for spatial relations often are based on concepts from small-scale space, and are metaphorically extended to large-scale (geographic) space. Reference frames, which form an important basis both for spatial language and for spatial reasoning, are discussed. Lastly, we set as a short term but important goal a search for geometries of spatial language.

INTRODUCTION

Spatial relations do not exist in the real world; rather, they exist in minds, to aid in making sense of that world, and in interacting with it. This paper discusses two approaches to the definition of spatial relations: experiential and formal. *Experiential models* are based on sensorimotor and visual experiences with the environment. Since it appears that most people experience the world in similar ways, experiential models of geographic space are expected to have much in common across individuals; spatial properties and relations in experiential models also should conform well with principles of naive (or common-sense) physics. Experiential models of space reveal themselves through spatial reference in natural language and through spatial behavior, either natural or under experimental conditions. On the other hand, *formal models* of geographic space employ mathematical or logical axioms and principles to build formal geometries, topologies, algebras, and logics for representing and manipulating spatial relations and objects. They may bear strong similarities with experiential models because often they have been developed to deal with the same kinds of properties of human observation and experience. For example, 'geometry' is said to have begun as rules and procedures used for land survey in ancient Egypt; Euclid further formalized these principles. Euclidean geometry is closely related to Newtonian (solid-body) physics; however, Newtonian physics itself corresponds closely with naive physics in many (but not all) every-day situations. *Experiential realism*, a philosophical basis for cognitive science advanced by George Lakoff (1987) and Mark Johnson (1987), and discussed recently in a geographic context by Couclelis (1988), is central to the models discussed here.

One of the five high-priority topics for research by the National Center for Geographic Information and Analysis (NCGIA) is "a general theory of spatial relationships" (Abler, 1987, p. 304). Abler goes on to elaborate that the goal is "a coherent, mathematical theory of spatial relationships" (Abler, 1987, p. 306). On the same page, he also states:

"Fundamental spatial concepts have not been formalized mathematically and elegantly. Cardinal directions are relative concepts, as are ideas basic to geography such as near, far, touching, adjacent, left of, right of, inside, outside, above, below, upon, and beneath."

A "theory of spatial relationships" should not only be mathematically elegant. Its concepts also must correspond with those concepts used by human minds during spatial cognition, spatial reasoning, and spatial behavior; otherwise, it will be of little if any use to

geographers, spatial analysts, or geographic information systems (GIS) users. Thus the search for "fundamental spatial concepts" must be conducted in the cognitive sciences before or in parallel with searches in mathematics.

Of course, this search for fundamental spatial concepts as a basis for geographic data structures is not new. More than a decade ago, several papers at the Harvard symposium on data structures for GIS addressed just these issues, and provided a number of approaches (in particular, see Chrisman, 1979; Kuipers, 1979; Sinton, 1979; Youngman, 1979). Now, however, the emergence of cognitive science, which seeks formal representations of how the mind deals with various phenomena, provides a new basis for advancing the topic.

In this paper we expand on the concepts and assertions mentioned above, and propose a strategy for relating various models of geographic space and concepts of fundamental spatial relations. We use spatial language, i.e., the terms in human language that people use to refer to spatial situations, as an important indicator of the major ways in which people conceptualize space. This is in some contrast to the approach used recently by Peuquet (1988), who based her "conceptual synthesis" for representations of geographic space more strongly on models of vision. An important goal of our approach is to identify those spatial concepts that are invariant under groups of transformations; this should contribute substantially to mathematical studies in both cognitive science and geography. This paper is a preliminary report on work in progress by both authors. We hope to reach a more comprehensive understanding of these topics ourselves, but also believe that some of the questions posed here will be of interest to others. Some of the material presented in this paper is taken from drafts of other manuscripts which we plan to publish in the near future.

MODELS OF 'SMALL-SCALE' SPACE

Downs and Stea (1977, p. 197) distinguished perceptual space, studied by psychologists such as Jean Piaget and his colleagues and followers, from "transperceptual" space that geographers deal with and that we are discussing in this paper. They claimed that "the two scales of space are quite distinct" (p. 197) in the ways people perceive and think about them. Later in the book, Downs and Stea (p. 199) contrasted the terms "small-scale perceptual space" and "large-scale geographic space." At about the same time, Benjamin Kuipers (1978, p. 129) defined *large-scale space* as "space whose structure cannot be observed from a single viewpoint." The large-scale/ small-scale distinction of Kuipers does not quite correspond to a geographic/non-geographic contrast, since as Kuipers pointed out, a high

mountain viewpoint or an aircraft permits direct visual perception of fairly large areas. Nevertheless, we will follow Kuipers, and use the term *large-scale space* as he defined it, and *small-scale space* to refer to subsets of space which are visible from a single point.

Our cognitive models of small-scale space develop from direct perceptions of our everyday world, dominated by a combination of visual inputs and the interactions of our bodies with the objects in that space. People are very good at processing the visual field, and at interpreting observed sequences of two-dimensional images to be views of objects in a three-dimensional space; in fact, it has been claimed that "the visual system attempts to interpret all stimulation reaching the eyes as if it were reflected from a scene in three dimensions" (Haber and Wilkinson, 1982, p. 25). Michael Crichton describes the relation between visual inputs and the geometry of small-scale space: "When you move inside a space, you must consciously be registering the distortions of the shapes, the moving walls, and corners. Only you don't interpret these as changes in the room itself, but use them as more accurate cues to orient yourself in the space" (Minsky, 1986, p. 256).

As noted above, bodily (sensorimotor) experiences with small-scale space also play a key role in the ways we build our mental models of such spaces. Lakoff and Johnson (Lakoff and Johnson, 1980; Lakoff, 1987; Johnson, 1987) claim that our spatial concepts for small-scale space largely are projected from human-body space. The ways in which the body interacts with objects allow us to recognize 'basic-level' objects such as 'chairs' by the age of about two years (see Rosch, 1973); many spatial-relational words are derived from body parts (for a recent review, see Svorou, 1988).

People also build cognitive models of the way familiar objects behave (react to forces) in small-scale space. The field known as *naive physics* (sometimes 'common-sense physics') deals with the ways in which people typically *think* that physical objects behave. For example, many people not trained in formal physics think that, when a person drops a ball while walking, the ball will fall straight down (McClosky, 1983). In an experiment described by McClosky (1983, p. 125), 80% of college student with no physics training, and 27% of those who had completed at least one physics course dropped a golf ball directly over the target. Of course, according to formal physics, the ball retains a forward motion component, falls in a parabola, and must be dropped before the hand is directly over a target in order to hit that target. Naive physics has associated with it concepts of distance, direction, connectivity, continuity, etc., which might be termed a 'naive geometry'.

Perception of the physics of everyday objects, together with our own bodily structures, also influences the way we perceive and label the structure of space. Gravity is so pervasive that the up-down axis is obviously the most *salient*, or most important to human perception and cognition. The horizontal plane, perpendicular to this vertical axis, is less clearly differentiated in the environment. However, for humans, the front-back contrast, while less salient than up-down, is considerably more salient than left-right. This observation, discussed by Freeman (1975) and many others, probably arises due to the fact that humans and most other animals show bilateral symmetry for external and most internal components. This salience ordering of the three dimensions of everyday space (up-down >> forward-back >> left-right), and the fact that the latter distinction is necessarily egocentric, is important to the models discussed later in this paper.

Introduction of concepts of measurement, mathematics, and science, especially during the time of the classic Greek philosophers, required that geometry and physics be formalized. Schoolbooks tell us that plane geometry was first formalized in Egypt to allow for land-ownership boundaries (the cadastre) to be re-established after the annual floods of the Nile. Abstraction of this practical formalization into a set of axioms is often credited to Euclid. Euclidean geometry conforms by and large to the naive geometry which we observe in our everyday lives. Current school curricula instill upon the pupil the idea that Euclidean geometry is the only 'correct' geometry.

A formal theory of physics proved more elusive, and Aristotelean physics is known to be fundamentally flawed (see Di Sessa, 1982, for a discussion of Aristotelean, Newtonian, and naive physics). The formal physics which corresponds closely to the behavior of everyday objects in small-scale space is usually attributed to Sir Isaac Newton. Newtonian (solid-body) physics is consistent with Euclidean geometry, and corresponds with naive physics well enough that people who 'believe in' Newtonian physics can deal with everyday objects as if the objects were governed by its 'Laws'. (For discussions of naive physics, see McClosky, 1983, or Hobbs and Moore, 1985.) Newtonian physics conforms closely with observable reality, while at the same time is a highly abstract, formal system which is extremely useful in engineering and scientific applications, where it can be used to build models and to predict accurately the behavior of mechanical systems.

MODELS OF 'LARGE-SCALE' ('GEOGRAPHIC') SPACE

The region of space that we can experience bodily at any moment is limited to a few cubic meters; the region we can experience visually usually is larger and much more variable, but still generally is much

smaller than the combined extent of all the spaces that we experience during the course of a day's activities. Benjamin Kuipers' model of spatial knowledge acquisition (Kuipers, 1978, 1983a, 1983b) begins from a sensorimotor experiential base. As we move through large-scale space, we see a sequence of views (a 'view' is defined as the sum total of all sensory inputs when at a point and oriented in a particular way, but for most people, the 'views' are dominated by visual inputs). With some views, we associate actions; some actions form part of the navigation or way-finding process, and other actions relate to other activities. Kuipers' TOUR model (implemented in LISP) uses as input ordered sequences of view-action (V->A) pairs. The routes form a 'spaghetti' of familiar paths, which constitute procedures for getting from one place to another. (Interestingly, although we first used this metaphor because of the frequent use of the term 'spaghetti files' in digital cartography, Bruce Chatwin [1988, p. 16] explicitly used the 'spaghetti' metaphor in describing the models of geographic space that are central to Australian aboriginals' myths and traditions; "One should perhaps visualize the Songlines as a spaghetti of Iliads and Odysseys, writhing this way and that, in which every 'episode' was readable in terms of geology.") Many other mobile organisms presumably have similar internal representations of large-scale space. Note that this kind of spatial knowledge is termed 'topological' by Piaget and his followers (Piaget and Inhelder, 1956), and 'procedural' by Thorndyke and Hayes-Roth (1982) and by Mark and McGranaghan (1986).

Kuipers (1978, 1983a, 1983b) noted that, as people find their way along various paths, they may recognize that the paths have some points ('places') in common. This allows them to use inference rules to build network models of places and connections, paths and barriers, in large-scale space. Such a cognitive model of large-scale space allows route-planning to novel destinations, or the planning of alternate routes when habitual paths are blocked. (Incidentally, such adaptive route-planning appears to not be restricted to human beings; Tolman (1948) discussed experiments in which laboratory rats were observed to use alternate paths when the usual ones were blocked by barriers.) Paths may have associated with them properties such as length in miles, kilometers, or blocks, or expected traversal times, but global geometric properties, such as coordinate locations, straight line ('as the crow flies') directions and distances between points, etc., often are weakly defined, inaccurate, or are absent from the model. Such properties of some cognitive models of large scale space were noted very early by Trowbridge (1913).

In Kuipers' TOUR model, spatial inference rules allow the model to be refined more and more, as more and more (V->A)-pair sequences are learned and assimilated, until a 'geometrically-correct' model of large-scale space is built up. However, it seems that, for many

people, such a two-dimensional Euclidean (cartesian) model of large-scale space is never built from experience alone, or at least that it takes a very long time. Mark and McGranaghan (1986, p. 402) felt that "access to graphic, metrically-correct maps almost certainly plays a key role" in the development of a cartesian cognitive model of geographic space. Such a conjecture is implicit in the findings of Thorndyke and Hayes-Roth (1982), and is supported by recent experiments by Lloyd (1988).

In his presentation at Auto Carto 8, Matthew McGranaghan stated that the power of maps comes from the fact that they represent space with space. In fact, maps represents *large-scale* (geographic) space in a *small-scale* space on a piece of paper or a computer screen, allowing us to 'vicariously experience' the geometry of the large-scale space in a 'familiar' way, that is, in the way we experience small-scale space (such as objects on a desk-top) in our everyday lives. Thus the map allows us to extend Euclidean geometry (which is a very good approximation to the 'true' or 'objective' geometry of small-scale space) outward onto large-scale space, to be used as a basis for certain forms of spatial inference, reasoning, and decision-making.

There is little doubt that maps do allow people to extend a model of the geometry of small-scale space outward to large-scale space; however, this may be judged to be 'good' or 'bad', depending on beliefs about 'truth', or on the uses to which the model of large-scale space is to be put. If one believes that Euclidean geometry is also the 'true' or 'objective' geometry of large-scale space, then the map is a very valuable tool, since it allows us to grasp this 'truth' and use it. With a map in hand, or with a map-based cognitive model of space, we can plan routes and perform other spatial inference using the familiar Euclidean model. However, if the 'true' geometry of large-scale space is believed to be the type or level of cognitive map which is acquired only through direct experience (and such experiential cognitive models almost certainly are not Euclidean), then the fact that maps extend small-scale geometric principles to large-scale space means that they are an 'incorrect' model for large-scale space. As early as 1980, Drew McDermott argued that the topological view of space inherent in Kuipers' model is not a good theoretical basis for spatial reasoning (McDermott, 1980, p. 246). As an alternative, McDermott proposed a theory of "metric spatial inference" based on a "fuzzy map" geometry, of positional uncertainty within a Euclidean coordinate framework. Later, McDermott and Davis (1984, p. 107) proposed an intermediate or hybrid model, in which the cognitive map might "consist of an assertional data base for topological information and a 'fuzzy map' for the metric information."

WHAT IS THE 'OBJECTIVE' GEOMETRY OF GEOGRAPHIC (LARGE-SCALE) SPACE?

What is meant by 'correct' geometry? We begin with the assumption that the 'real world' exists, and that it has 'objective' properties. This is an assumption and not a 'fact', since the human mind has no 'direct' access to the real world, but only is aware of what the senses appear to report. Since the decision to adopt a particular definition of objectivity is itself subjective, Hillary Putnam has shown that a paradigm of complete objectivity is internally inconsistent (see discussion in Lakoff, 1986, pp. 229-259). Nevertheless, *experiential realism*, proposed by Lakoff and Johnson (1980) under the term *experientialism*, is based on the idea that there *is* a real world, which has consistent properties, so that when people interact with that world, their mental experiences are very similar (see Lakoff, 1987, especially pp. 265-268). If we adopted *reproduceability in measurement* as an essential part of our definition of 'objective' reality, then the 'correct' geometry is the one which best supports reproducible measurement of positions in real-world geographic space, namely, 'the' geometry of surveying. At scales ranging from planet earth to the human body, Euclidean geometry and Newtonian physics seem quite adequate. The fact that Euclidean geometry breaks down at certain time, space, or velocity scales, and that Einstein's theory of relativity required new geometries, thus re-orienting the cutting edge of academic geometry, is of little relevance to geography and surveying.

Even if Euclidean geometry is 'correct' in the narrow ('objective') sense stated above, it still does not seem to represent large-scale space the way most people think about it, or the way in which they reason while way-finding in a familiar large-scale space. However, map-based reasoning, that is, spatial reasoning based on a Euclidean two-dimensional geometry, may well be the best available form of spatial reasoning for navigation and other spatial tasks when those tasks must be performed *in an unfamiliar environment*. The high annual sales figures for road maps and road atlases support the idea that most non-technical people believe that maps are, if not optimal, at least very good in this regard.

It is not far wrong to view our planet as a spheroidal solid body in Euclidean 3-space; geodesy has established the shape of that body, and of the geoid. The surface of the earth is essentially a 2-dimensional manifold stretching over the surface of that geoid; position can be denoted as two angles (latitude and longitude), and elevation above 'sea-level' at any point may be defined as the height above that geoid. Map projections allow us to transform from one 2-dimensional surface (over the spheroid) to another (a cartesian

plane) in ways which control the geometric distortions that necessarily result.

For 'sufficiently-small' regions of the planet (say, up to about the size of the 48 contiguous United States, or Australia), the curvature of the planet can more or less be ignored; map projections exist which show almost no distortion of areas, angles, or distances over regions of that size or smaller. For example, Lambert's conformal conic projection with standard parallels at 33 and 45 degrees north provides correct angles (by definition), and a maximum scale-variation of one-half of one percent between latitudes 30.5 and 47.5 over the 48 contiguous United States (Snyder, 1982). Thus it is 'reasonable' to treat the cartesian coordinates of Lambert's projection as the basis for a Euclidean view of the geographical geometry of the contiguous United States, or of subregions thereof.

Measurement is often considered to be the only way to 'see' space in an objective way (because it is obviously reproducible). However, it also is possible to define 'correct' in a way which does not to rely on the concept of measurement. People usually experience space not by measurements, but rather by observing results of processes that are related to space. An every-day examples for such a process is that time is consumed by physical movement in space, and then that there are other 'costs' of travel. This approach also is applied to observations of how social systems behave in space, including the perception and cognition of regions and urban centers.

This experience with processes that are influenced by other properties of geographic space creates another, indirect concept of space that is found among neither the concepts learned through navigation in large-scale space nor through the utilization of concepts from small-scale space to organize spatial precepts. To a degree that these processes avail themselves to 'objective' measurement, the spatial properties they react to can be indirectly observed and deduced. On a conceptual level, the difficult task is to combine the multiple, conflicting concepts human beings use and understand in their interaction, and to model how they influence specific behavior. Geography deals with many of these spatial processes, and thus geography and geographers can play a key role in discovering the spatial properties influencing these processes; this may in turn help to understand human spatial cognition.

SPATIAL RELATIONS

John Freeman (1975) provided an important and early review paper on formal representation of spatial relations. Freeman proposed that the following form a complete set of primitive spatial relations for elements in a (2D) picture, a view of a (3D) small-scale space: 1. left

of; 2. right of; 3. beside (alongside, next to); 4. above (over, higher than, on to of); 5. below (under, underneath, lower than); 6. behind (in back of); 7. in front of; 8. near (close to, next to); 9. far; 10. touching; 11. between; 12. inside (within); and 13. outside. Note that this is not a *minimal* set of relations, since some can be defined as combinations of some of the others.

Freeman's list is very similar to the list of terms presented by Abler (1987, p. 306) and quoted in the introduction to this paper. The cardinal directions can be added to Freeman's list through the addition of one more axiom. If we associate 'north' with 'up', then 'south=down', 'west=left', and 'east=right' can follow deductively. Peuquet and Zhan (1987) extended Freeman's (1975) relation set in exactly this way, including the cardinal directions as spatial relations without comment, and substituted 'north' for 'above' and 'south' for 'below' in the example they drew from Freeman's paper (Peuquet and Zhan, 1987, p. 66). Note that the 'north=up' axiom is quite arbitrary. Indeed, the etymology of the Indo-European root for the word 'north' is based on 'left' (Svorou, 1988); this relation results from an earlier 'east=forward' convention, and world maps in Medieval times were presented with an east up *orientation* (*orient=east*).

It is useful in such a system to maintain a strong society-wide tradition of keeping the same cardinal direction 'up' in both mapping and speech; although there are many local exceptions (usually based on locally salient physical gradients; see Mark, Svorou, and Zubin, in press), using 'up' to refer to directions other than north is considered to be 'wrong' by many English-speakers with little or no formal cartographic or geographic training. Note that in this model, the north-south geographic axis is 'bound' to the most salient directional axis (up-down) of small-scale space, and (east-west) is bound to the least salient of these (left-right). Could this account for the tendency for some people to confuse east-west more readily than north-south?

Some cultural and linguistic groups, including the Hawaiians and many other island dwellers, use a radial coordinate system for referencing in large-scale space (see Mark and others, in press). This uses the 'inside-outside' dichotomy of the container metaphor (see Lakoff, 1987) for one spatial dimension, and 'toward some landmark' (spatial action, rather than relation) as the other. Other island peoples use similar spatial reference frames (see Haugen, 1957, for a discussion of this for Iceland).

REFERENCE FRAMES

Mark and others (in press) have noted the importance of reference frames in the generation of spatial language. For example, a building such as a church has around it a region or ground. The structure of the church and the ways people interact with it give the church a 'front', a 'back', sides, *et cetera*. Then, these parts project outward onto adjacent regions of the ground, leading to the division of that ground into subregions: the subregion adjacent to the back of the church can be referred to as 'behind the church', and so on. The church and its region provide a reference frame for spatial language comprehension and generation. McDermott's (1980) approach to spatial inference based on fuzzy maps also uses reference frames as a central concept:

"All of our solutions revolve around keeping track of the *fuzzy coordinates* of objects in various *frames of reference*. That is, to store metric facts about objects, the system tries to find, for each object, the ranges in which qualities like its X and Y coordinates, orientation and dimensions lie, with respect to convenient coordinate systems. The set of all the frames and coordinates is called a *fuzzy map*. We represent shapes as prototypes plus modifications." (McDermott, 1980, p. 246).

If the model get new facts that do not reduce uncertainty, they add features to the model. McDermott gives as an example within the Yale University campus database "the orientation of Sterling Library is the same as the orientation of Becton Library". This adds to the database a new reference frame F, within which (ORIENT STERLING) = (ORIENT BECTON) = 0.0. Then the frame F is itself a new object, with orientation completely fuzzy (0, 2Pi) with respect to other features of the Yale campus.

"Every object can serve as a frame of reference, and every frame of reference can be considered an object, with a position, orientation, and scale" [within some parent frame of reference]. (McDermott, 1980, p. 247).

In McDermott's model, the nested frames form a tree, which can be rearranged as new facts are added. Cognitive hierarchies of reference frames may not be so simple, and may be networks with loops, multiple hierarchies, *et cetera*.

McDermott has used this as a basis for spatial inference, and for building within the machine a 'cognitive map' (see also McDermott and Davis, 1984). However, as noted above, reference frames form an important basis for the interpretation and generation of spatial language. Furthermore, Roger Downs has suggested that the concept

of spatial hierarchy is of critical importance to spatial knowledge acquisition (see Downs' section in Mark, 1988, pp. 5-6). Inference based on hierarchy appears to be very important in much of everyday spatial decision making, and also 'accounts for' the surprise that Reno Nevada is west of San Diego, or that Atlanta is closer to Chicago than it is to Miami. Stewart Fotheringham (personal communication, 1988) is examining residential choice in this context.

THE RELATION 'NEAR'

The word 'near' embodies a fundamental spatial relationship that applies to object pairs in geographic as well as in other spaces. It is among both Freeman's (1975) and Abler's (1987) lists of fundamental spatial relations. Robinson and his co-workers (Robinson and other, 1985, 1986; Robinson and Wong, 1987) have studied the meaning of 'near' from the point of view of fuzzy sets.

Mark Johnson recognizes the importance of 'near' in his discussion of how image schemata, and in particular the center-periphery schema, constrain meaning, understanding, and rationality:

"Given a center and a periphery we will experience the NEAR-FAR schema as stretching out along our perceptual or conceptual perspective. What is considered near will depend upon the context, but, once that is established, a SCALE is defined for determining relative nearness to the center."
(Johnson, 1987, p. 125) .

Mark and others (in press) also recognized the scale-dependant nature of the meaning of 'near'. As an example, the statements: "Santa Barbara is near Los Angeles", "My house is near the University", and "My barbecue is near my swimming pool", all sound reasonable, although the ranges of inter-object distances involved are clearly very different. How is it that the listener or the reader 'knows' what the above assertions mean? In fact, Robinson's results show that even when context, object class, and universe are held constant, there still are individual variations in the meaning of 'near.'

Prototypes form an important part of Lakoff and Johnson's "experiential realism" model of cognition (Lakoff, 1987; Johnson, 1987). As an example given by Lakoff (1987), a 'small galaxy' is not an object in the intersection of 'the set of all galaxies' and 'the set of small things'; we know that the phrase means, "of the sizes that galaxies come in, this particular one is smaller than most." A set-theoretic model, even a fuzzy set model, would have trouble representing the fact that a 'small galaxy' is many orders of magnitude larger than a 'large mouse'. Here, we propose the

conjecture that 'near' is a similar concept, and that it takes its meaning from prototypical distances *or interactions* between the kinds of objects in the statement. If this is correct, the problem of determining the meaning of 'near' must begin by determining, from the kinds of objects involved and other aspects of context, what the appropriate prototype is. This is a research topic of high priority.

For someone who knows the 'driving' culture of southern California, the context for "Santa Barbara is near Los Angeles" is inter-city travel by private automobile, using freeways or other highways (with speed limits of around 90 km/hour). The sentence "Santa Barbara is near Los Angeles" might easily be misunderstood by someone from outside North America who has never owned a car. If the listener knows that the speaker works at 'the University', then the context for "My house is near the University" probably lies in what geographers call 'journey-to-work', whereas the context for "My barbecue is near my swimming pool" lies in typical layouts of backyard furniture and appliances. Note that prototypes for the first two situations will be based primarily on spatial interaction; only the last situation has a more static prototype.

The 'near' relation is considered especially critical because it is implicit in many other spatial relations. For example, consider the question of how close together two objects must be in order that the expression "a is in front of b" makes sense. If someone states: "Your bicycle is in front of the house", you would not expect it to be 7 kilometers from the house, even if a straight line from the bicycle to the house meets the front of the house at right angles. In at least most cases, it seems that "A is in front of B" means something like "'A is in front of B' and 'A is near B'".

TOWARD A GEOMETRY OF LANGUAGE

It is clear that cognitive models of small-scale and large-scale space are related to the language that people use to describe and communicate about such spaces. How shall these research areas be linked in formal models? We propose that the development of a 'geometry of language' would be a major step in the direction of such an integration.

After the invention of non-Euclidian geometry, the term geometry needed a new definition, that would include study areas such as 'graph theory' and other more general, but somewhat geometrical topics. Felix Klein [Klein-Erlanger] defines geometry as the science concerned with properties of figures which are invariant under a group of transformations. Transformation is here understood as a mapping of the space

onto itself, thus including not only the familiar transformations such as translation and rotation but also many others.

Let us first explore this very general definition informally. Assume a space and a figure (i.e. a subset of the space) in it. The notion of space here is more primitive than the one used previously [in Frank 1988] (otherwise we would be caught in a circular definition). It is essentially a finite or infinite collection of discernable objects (typically the points in the space) together with a notion of 'neighborhood'. Then, geometry deals with properties of these figures (eg. lengths of lines, connections between points) which remain unchanged under a class of transformations (eg. translations, rotations, map projections). Each group of transformation defines another set of geometric properties (distance, angle, area). This notion seems to capture more of what people understand by 'geometry' than is included in classical Euclidian geometry with its points and line. [adapted from Frank 1988].

Couclelis and Gale (1986) also have discussed spatial concepts from a basis of algebraic group theory.

As noted above, a geometry may be defined as "properties which remain invariant under a group of transformations". Thus, we might ask: "Is there a geometry of natural language?" That is, are there properties of the relations between spatial language and the real world which remain unchanged by certain geometric transformations? If so, what are the words or phrases, and what are the transformations? Containment is invariant under many transformations: "the building is inside the fence", "Buffalo is in New York state", and "his grave is in the mission cemetery" will remain true under a very wide range of spatial transformations. However, the statement: "The cemetery is north of the church" will be true if the church-cemetery pair is translated across geographic space, but will not be true after a rotation of 90 degrees. However, the similar statement: "The cemetery is *behind* the church" remains true under both translation and rotation, as long as the transformation is applied to a region including both objects.

Here we again see the critical nature of reference frames. Spatial language of the form "A is behind B" is invariant under a rotation *if and only if the reference frame rotates with the referent*. The cardinal directions are abstract and fixed with respect to the planet; we already ignore the orbiting, rotating, and other astronomical motions of the Earth. Thus, since the astronomical reference frame is never rotated, expressions "A is north of B" would not be invariant under large rotations of the region including both A and B. Since "the

cemetery" would almost always fall within the same ground and reference frame as "the church", it thus would rotate with it.

A different and interesting case would be the radial reference frame common to Hawaii and many other oceanic islands, already mentioned above. If the entire island were rotated, the meaning of the language would not change, but if a town or shopping center is rotated, the spatial language probably would no longer apply. In this case, since the reference frame is radial, a sufficiently large translation of a subregion of the island, with no rotation, could also make the utterance untrue. Reference frames will play a critical role in the development of any geometry of spatial language.

SUMMARY

Development of a comprehensive model of spatial relations and properties is important to the future of systems for geographic information and analysis, and also to cognitive science and to behavioral geography. This paper first reviewed concepts of space. A critical distinction is between small-scale spaces, whose geometry can be directly perceived through vision and other senses, and large-scale space, which can be perceived only in relatively small parts. Fundamental terms for spatial relations often are based on concepts from small-scale space, and are metaphorically extended to large-scale (geographic) space. Thus, terms and concepts for the spatial relations among the objects in a picture can form an appropriate core for spatial language. Spatial relations at a geographic scale are formed either by extension of these terms, or by addition of a small set of additional principles (for example, letting "north" equal "up"). Reference frames form an important basis both for spatial language and for spatial reasoning. Prototypes also are important, and probably play a central role in the way we determine the geometrical meaning of spatial relations such as 'near'. Finally, we set as a short term but important goal a search for geometries of spatial language. This search will attempt to define those properties of particular instances of spatial reference in natural language which remain invariant under groups of transformations. This could form the basis both for aspects of geographic data structures and for the understanding and generation of spatial language itself.

ACKNOWLEDGEMENTS

This paper represents part of Research Initiative #2, "Languages of Spatial Relations", of the National Center for Geographic Information and Analysis, supported by a grant from the National Science Foundation (SES-88-10917); support by NSF is gratefully

acknowledged. The paper was written in part while Mark was a Visiting Scientist with the CSIRO Centre for Spatial Information Systems, Canberra, Australia. Ronald Amundson, Sherry Amundson, and Matthew McGranaghan provided useful comments on earlier drafts of this paper.

REFERENCES

- Abler, Ronald F., 1987. The National Science Foundation National Center for Geographic Information and Analysis. *International Journal of Geographical Information Systems*, v. 1, no. 4, 303-326.
- Chatwin, Bruce, 1988. *The Songlines*. London: Pan Books Ltd., 327.
- Chrisman, Nicholas, 1979. Concepts of space as a guide to cartographic data structures. In Dutton, G., editor, *Proceedings, First International Study Symposium on Topological Data Structures for Geographic Information Systems*, Volume 7 ("Spatial Semantics: Understanding and Interacting with Map Data"), pp. CHRISMAN/1-CHRISMAN/19.
- Couclelis, Helen, 1988. The truth seekers: Geographers in search of the human world. In Golledge, R., Couclelis, H., and Gould, P, editors, *A Ground for Common Search*. Santa Barbara, CA: The Santa Barbara Geographical Press, pp. 148-155.
- Couclelis, Helen, and Gale, Nathan, 1986. Space and spaces. *Geografiska Annaler*, 68B, 1-12.
- Di Sessa, Andrea, 1982. Unlearning Aristotelean physics: A study of knowledge-based learning. *Artificial Intelligence* 6: 37-75.
- Downs, Roger M., and Stea, David, 1977. *Maps in Minds: Reflections on Cognitive Mapping*. New York: Harper and Row.
- Frank, Andrew U., 1988. The concept of space and geometry in geography. Draft manuscript, Department of Surveying Engineering, University of Maine, July 13 1988, 31pp.
- Freeman, John, 1975. The modelling of spatial relations. *Computer Graphics and Image Processing* 4: 156-171.
- Haber, Ralph Norman, and Wilkinson, Leland, 1982. Perceptual components of computer displays. *IEEE Computer Graphics & Applications*, May 1982, 23-35.
- Haugen, E., 1957. The semantics of Icelandic orientation. *Word* 13: 447-459.
- Hobbs, Jerry R., and Moore, Robert C., 1985. *Formal Theories of the Commonsense World*. New Jersey: Ablex.
- Johnson, Mark, 1987. *The Body in the Mind: The Bodily Basis of Meaning, Imagination, and Reason*. Chicago: University of Chicago Press.
- Kuipers, Benjamin, 1978, Modeling spatial knowledge. *Cognitive Science* 2, 129-153.

- Kuipers, Benjamin, 1979. Cognitive modelling of the map user. *In* Dutton, G., editor, *Proceedings, First International Study Symposium on Topological Data Structures for Geographic Information Systems*, Volume 7 ("Spatial Semantics: Understanding and Interacting with Map Data"), pp. KUIPERS/1-KUIPERS/11.
- Kuipers, Benjamin, 1983a. The cognitive map: Could it have been any other way? *In* H. L. Pick, Jr., and L. P. Acredolo, editors, *Spatial Orientation: Theory, Research, and Application*. New York: Plenum Press, pp. 345-359.
- Kuipers, Benjamin, 1983b. Modeling human knowledge of routes: Partial knowledge and individual variation. *Proceedings, AAAI 1983 Conference*, The National Conference on Artificial Intelligence, pp. 1-4.
- Lakoff, George, 1987. *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind*. Chicago: University of Chicago Press.
- Lakoff, George, and Johnson, Mark, 1980. *Metaphors We Live By*. Chicago: University of Chicago Press.
- Lloyd, Robert, 1988. Information and cognitive maps [abstract only]. Paper presented to the Annual Meeting of the Association of American Geographers, Phoenix, Arizona, April 6-10, 1988, abstract published in *1988 AAG Annual Meeting Program and Abstracts*, p. 113.
- Mark, David M., (editor) 1988. *Cognitive and Linguistic Aspects of Geographic Space: Report on a Workshop*. Santa Barbara, CA: National Center for Geographic Information and Analysis, Technical Report 88-6.
- Mark, David M., and McGranaghan, Matthew, 1986. Effective provision of navigation assistance to drivers: A cognitive science approach. *Proceedings, Auto Carto London*, vol. 2, 399-408.
- Mark, David M., Svorou, Soteria, and Zubin, David A., in press. Spatial terms and spatial concepts: Geographic, cognitive, and linguistic perspectives. *Proceedings, International Symposium on Geographic Information Systems: The Research Agenda*. Crystal City, Virginia, November, 1987, proceedings in press.
- McClosky, Michael, 1983. Intuitive physics. *Scientific American* 248 (4), 122-130, April 1983.
- McDermott, Drew, 1980. A theory of metric spatial inference. *Proceedings, First Annual National Conference on Artificial Intelligence* (American Association for Artificial Intelligence), 246-248.
- McDermott, Drew, and Davis, Ernest, 1984. Planning routes through uncertain territory. *Artificial Intelligence*, 22, 107-156.
- Minsky, Marvin L., 1986. *The Society of Mind*. New York: Simon & Schuster, 339 pp.

- Peuquet, Donna J., 1986. The use of spatial relationships to aid spatial database retrieval. *Proceedings, Second International Symposium on Spatial Data Handling*, Seattle, Washington, 459-471.
- Peuquet, Donna J., 1988. Representations of geographic space: toward a conceptual synthesis. *Annals of the Association of American Geographers*, 78, 375-394.
- Peuquet, Donna J., and Zhan Ci-Xiang, 1987. An algorithm to determine the directional relationship between arbitrarily-shaped polygons in the plane. *Pattern Recognition*, 20, 65-74.
- Piaget, Jean, and Inhelder, B., 1956. *The Child's Conception of Space*. London: Routledge & Kegan Paul.
- Pullar, David V., and Egenhofer, Max J., 1988. Toward formal definitions of topological relations among spatial objects. *Proceedings, Third International Symposium on Spatial Data Handling*, Sydney, Australia, August 17-19, 1988, pp. 225-241.
- Robinson, Vincent B., Blaze, M., and Thongs, D., 1986. Representation and acquisition of a natural language relation for spatial information retrieval. *Proceedings, Second International Symposium on Spatial Data Handling*, Seattle, Washington, July 1986, pp. 472-487.
- Robinson, Vincent B., Thongs, D., and Blaze, M., 1985. Machine acquisition and representation of natural language concepts for geographic information retrieval. *Modeling and Simulation* (Proceedings, 16th Annual Pittsburgh Conference), v. 16, Part 1, pp. 161-166.
- Robinson, Vincent B., and Wong, R., 1987. Acquiring approximate representation of some spatial relations. *Proceedings, Auto-Carto 8*, pp. 604-622.
- Rosch, Eleanor, 1973. On the internal structure of perceptual and semantic categories. In T. E. Moore (editor), *Cognitive Development and the Acquisition of Language*. New York, Academic Press.
- Sinton, David, 1979. The inherent structure of information as a constraint to analysis: Mapped thematic data as a case study. In Dutton, G., editor, *Proceedings, First International Study Symposium on Topological Data Structures for Geographic Information Systems*, Volume 7 ("Spatial Semantics: Understanding and Interacting with Map Data"), pp. SINTON/1-SINTON/17.
- Snyder, John P., 1982. Map projections used by the U.S. Geological Survey. *Geological Survey Bulletin* 1532, Second Edition, 313pp.
- Svorou, Soteria, 1988. *The Experiential Basis of the Grammar of Space: Evidence from the Languages of the World*. Unpublished Ph.D. Dissertation, Department of Linguistics, State University of New York at Buffalo.
- Thorndyke, Perry W., and Hayes-Roth, B., 1982. Differences in spatial knowledge acquired from maps and navigation. *Cognitive Psychology* 14, 560-589.

- Tolman, 1948. Cognitive maps in rats and men. *Psychological Review* 55, 189-208.
- Trowbridge, C. C., 1913. On fundamental methods of orientation and imaginary maps. *Science* 38, 888-897.
- Youngman, Carl, 1979. A linguistic approach to map description. In Dutton, G., editor, *Proceedings, First International Study Symposium on Topological Data Structures for Geographic Information Systems*, Volume 7 ("Spatial Semantics: Understanding and Interacting with Map Data"), pp. YOUNGMAN/1- YOUNGMAN/19.

**GEOGRAPHIC INFORMATION:
Aspects of Phenomenology and Cognition**

Major R.J. Williams
Department of Geography and Oceanography
Australian Defence Force Academy
CAMPBELL ACT 2600
AUSTRALIA

ABSTRACT

The disciplines of geography and cartography have experienced an on-going debate on the importance of topological structure in geographic data for the past two decades. It is only now that many organizations are realizing the importance of such structures. Therefore, having crossed the 'topology hurdle', the current trend in research is towards 'integration' of various data types and the 'object-orientation' of geographic features.

While these concepts of 'integration' and 'object-orientation' are applaudible, the theoretical approach seems to have some deficiencies. It appears that the aim of many researchers is to 'force' all geographic data into one basic structure, or another, such as 'vector-based' or 'raster-based'. Such an approach creates unnatural and illogical constructs of 'real-world' geographic phenomenon.

In response to this dilemma, this paper discusses phenomenological structures of geographic information and aspects of interpretation. Fundamental to this approach is the knowledge representation of phenomena of the 'real-world' independent of any specific application, and that analysis is based on actual geographic structure and location and not on graphical representations of that data.

INTRODUCTION

The rapidly growing population of the earth and the increasing complexity of modern life, with its attendant pressures and contentions for available resources, has made necessary detailed studies of the physical and social environment, ranging from population to pollution and from food production to energy resources and terrain evaluation. The geographer, preeminently, as well as the planner, historian, economist, agriculturist, geologist, military tactician, and others working in the basic sciences and engineering, long ago found the map to be an indispensable aid. With maps, the geographer may observe or record factual observations, describe the manner in which individual earth's phenomena vary from place to

place, develop hypotheses concerning the association of environmental factors, and, in general, study the spatial correlation of the elements of the earth's surface. By their very nature maps are the presentation of geographic spatial relationships (Robinson, Sale and Morrison, 1978; Trewartha, Robinson and Hammond, 1967).

Within the last decade, computers have been used with greater regularity to assist in the representation and analysis of geographic phenomena. The management and use of digital data has, however, been influenced by traditional approaches to mapping and resource management. These approaches to the management of data have, in turn, influenced how the geographic data has been structured. Three major classes of data structure have evolved; these being the *unlinked vector model* (used predominately within computer-assisted mapping systems), the *topological structured vector model* (used mainly in Land Information Systems, and with those types of data managed according to societal regulation); and the *grid cell and raster model* (generally used by natural resource managers, and with data obtained from remote sensing scanners).

Until recently these trends were often viewed as non-compatible approaches designed for different markets. But today's mapping market is seeing a major shift toward decision support and operations management. Supporters of the three trends, although continuing to foster their respective approaches, are now agreeing that there is a need to develop interfaces and integration between systems.

TOWARDS INTELLIGENT SYSTEMS

Up until now the *data models* have restricted the use of geographic data, but the major trend in research is to structure data suitable for multiple purposes and using *real-world* relationships (Grady, 1986). Bouillé (1986) asserts that "contrary of what is generally taught, we must emphasize the fact that the structure [of geographic data] is completely independent of the problem we want actually to solve. Moreover, a structure built correctly, and with no '*a priori*' idea, always contains a substructure which immediately answers our problem..." But, in order to achieve these desirable characteristics, far more intelligence has to be applied to data than in the past. The theory for these developments will come from the field of artificial intelligence, specifically the area known as *expert systems* (Williams, 1987)

Intelligence can be achieved via two fundamental, but integrated sources. These sources are those of data relationships and structure, and techniques and procedures for manipulating and analysing the data relationships. These sources can be considered as forming *expertise*. *Expertise* consists of *knowledge* about a particular domain (*real-world* geographic structures), understanding of the domain problems, and *skill* at solving some of these problems. *Knowledge* (in any speciality) is usually of two sorts: *public* and *private*. *Public*

knowledge includes the published definition, facts, and theories of which textbooks and references in the domain of study are typically composed. But *expertise* usually involves more than just this *public knowledge*. Human experts generally possess *private knowledge* that has not found its way into the published literature. This *private knowledge* consists largely of rules of thumb that have come to be called *heuristics*. *Heuristics* enable the human expert to make educated guesses when necessary, to recognize promising approaches to problems, and to check effectively with errorful or incomplete data. Elucidating and reproducing such *knowledge* is the central task in building *expert systems* (Hayes-Roth, Waterman, and Lenat, 1983).

The *phenomena of the real world*, (or data relationships and structure) can be described by an abstraction defined within functional categories, related via topological properties and spatially referenced. Data abstracted in such a way could be considered as being in a *geographic knowledge base*. The uses of this geographic data are affected by the requirements of a user's purpose. The products for this purpose can be static reports, such as map overlays of categorized and symbolized geographic data, or as a result of analyses or desired view of various aspects of the data. In a sense, these products result from processes by which information is selected and encoded, reduced or elaborated, stored and recovered, and decoded and used. Such products can be considered as having *cognitive* properties. Moore and Golledge (1976) suggest that "environmental cognition refers to awareness, attitudes, impressions, information, images, and beliefs that people have about environments. These environments may be directly experienced, learned about, or imagined. Environmental cognition refers to essentially large scale-environments, from nation and geographic regions down to cities and spaces between buildings, to both built and natural environments, and to the entire range of physical, social, cultural, political, and economic aspects of man's world. Cognition of these environments implies not only what individuals and groups have information and images about the existence of these environments and their constituent elements, but also that they have impressions about their character, function, dynamics and structural interrelatedness, and that they imbue them with meaning, significance and symbolic properties".

In summary, the *phenomena of the real world* are those abstractions which describe the earth's surface, its form and physical features, its natural and political divisions, the climate, the productions and the populations, whereas the *cognition* is the interpretation and analyses of relations between the phenomenological aspects.

THE PHENOMENA OF OUR DATA REALITY

Several researchers have developed and itemized "levels of data abstraction". Notable, within the field of geographic data description, there are contributions by Nyerges (1980) who identifies six levels of data abstraction and, more recently, work by Guptill et al (1987) who

identify five levels of abstraction. The important component within both schemes is the stated importance on the *data model* (or Nyerges' information and canonical structures).

Data Reality	The data existing as ideas about geographical entities and their relationships which knowledgeable persons would communicate with each other using any medium for communication
Information Structure	A formal model that specifies the information organization of a particular phenomenon. This structure acts as a skeleton to the canonical structure and includes entity sets plus the types of relationships which exist between those entity sets
Canonical Structure	A model of data which represents the inherent structure of that data and hence is independent of individual applications of the data and also of the hardware and software mechanisms which are employed in representing and using the data
Data Structure	A description elucidating the logical structure of data accessibility in the canonical structure. There are access paths which are dependent on explicit links and others which are independent of links
Storage Structure	An explicit statement of the nature of links expressed in terms of diagrams which represent cells, linked and contiguous lists, levels of storage medium, etc
Machine Encoding	A machine representation of data including the specification of addressing, data compression and machine code

Figure 1 Nyerges Levels of Data Abstraction

Spatial Data Model

The phenomena of data reality is considered, in totality, as *entities*. An entity and its digital representation is termed a *feature*. A feature is a set of phenomena with common attributes and relationships. All of the elements of this set of phenomena are homogeneous with respect to the set of selected common attributes and relationships used to define a feature. All geographic features implicitly have location as a defining attribute.

The concept of *feature* encompasses both entities and objects. The common attributes and relationships used to define the feature also apply to the corresponding entities and objects. An *entity* is a real world phenomena that is not subdivided into phenomena of the same kind. This 'real world phenomena' is defined by the attributes and relationships used to define the feature. An *object* is the representation of all or part of an entity. The concept *object*

encompasses both *feature object* and *spatial object*. A *feature object* is an element used to represent the non-spatial aspects of an entity. A *spatial object* is an element used to represent the position of an element (Moellering, 1987; Guptill et al, 1987; Rossmeissl et al, 1987).

An attribute is a characteristic of a feature, or of an attribute value. The characteristics of a feature include such concepts of shape, size, material composition, form and function of a feature. The attributes assigned to a feature include those inherent in the definition of the feature and additional attributes which further describe the feature. An attribute value is a measurement assigned to an attribute for a feature instance, or for another feature value.

Further, three major groups of rules can be used to formulate the description of feature instance. These groups are rules for defining feature instances, composition rules for representing feature instances, and rules for aggregating feature instances.

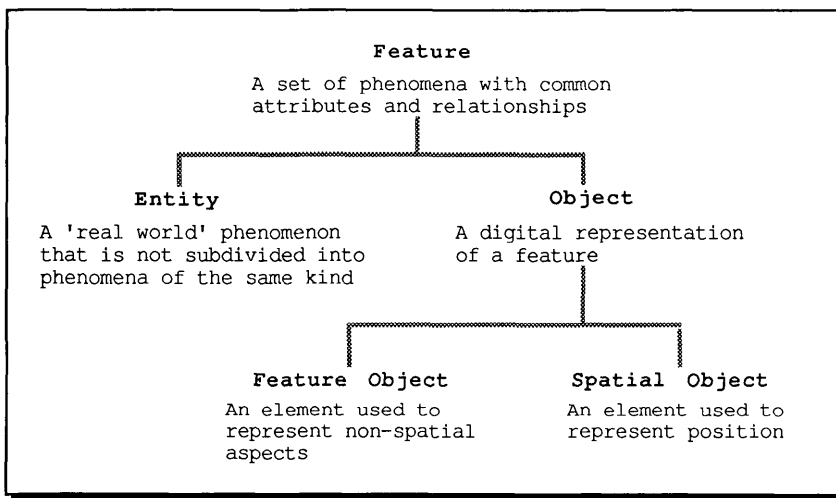


Figure 2 Concepts of feature, entity and object

World Views

Rossmeissl et al (1987) suggest that a methodology to define the domain of features is to use a concept of *World views* of spatial entities. Rossmeissl's proposed views are 'cover', 'division', 'ecosystem', 'geoposition', and 'morphology'. Each view is subdivided into subviews. An alternate approach would be to have *World views* related to functional categories as shown in Figure 3. The alternative approach is more applicable to *real-world* feature categories and corresponds to grouping commonly used in land inventories. Separate and independent research by McKeown (1983) indicates that *complex features* (spatial entities) can be represented using *schemas*. Each entity is a 'concept map' and can be represented by one

concept schema and at least one *role* schema. McKeown notes that using such an approach facilitates hierarchical decomposition of features, say, using natural hierarchies such as political boundaries, neighbourhoods, commercial and industrial areas, and so on (Figure 4).

Apart from categorizing the domain of features, aspects of perception (scale relevance) seem to be inadequately addressed by researchers. Rhind (1988) notes that 'existing computer geographic information systems (apart from Domesday) are entirely or very substantially based upon digital storage of coordinate data and their attributes - essentially low level conceptualizations of the objects under consideration'. He observes that "human beings evidently store multiple levels of conceptualization of objects, sometimes in a 'soft' or 'fuzzy' fashion". It seems that aspects of conceptualization can be defined using adaptations of Rossmeissl's 'views' and McKeown's 'concepts' to manage data from 'scale related' views.

Under the *expert system* approach, the aspects of *data reality* discussed in this section correspond to a formal definition of *public knowledge* and constitute a 'geographic knowledge base'. Having established this geographic knowledge base, it should be possible to extract information according to different interpretations aligned to the use of the information.

COGNITION OF OUR DATA REALITY

Traditionally, geographic data has been processed using one of three fundamental techniques. These are: (1) through graphic reports, such as standard series maps and thematic maps; (2) by overlay analysis for management and control of phenomena relevant to land management; and (3) using analytical techniques for planning and modeling of terrain related features (Figure 5).

These various interpretations can be produced by using combinations of existing data management and transformation techniques. As an example, standard series and thematic maps would use 'rules' for *world view* overlays and aggregation of features; techniques for generalization, simplification, and symbology; transformations for coordinate change; projection formulae, and processes for scale change, map derivation and update. Management and control products would require operators to process data that is represented with each object having spatial location as an essential property (*vector model*) and locations that have object properties (*tessellation or grid cell models*); techniques to overlay and integrate *world views* in both model types; and data base management functions. Planning and modeling products would require an extensive range of operators as mentioned already, as well as mathematical and topological functions; spatial data operators; and heuristic algorithms (Figure 6).

Regional Administration	Includes political, administrative, institutional, statistical facilities and regions; as well as reservations, parks, monuments, etc.
Population	Includes places of human habitation and occupation eg residential, commercial, religious, cultural, entertainment, recreational, educational, etc.
Road Infrastructure	Includes roads, junctions, bridges, overpasses, and related features, etc.
Rail Infrastructure	Includes lines, marshalling yards, bridges, and all related features, etc.
Air Infrastructure	Includes airfields, facilities, navigations aids, etc.
Sea Infrastructure	Includes port facilities, jetties, piers, sea control features, channels, canals, etc.
Telecommunications	Includes communication facilities, structures, networks, etc.
Electricity / Fuel	Includes power plants, facilities, and networks for generation and distribution of power and fuel.
Water Resources	Includes facilities and networks for storage and distribution of water resources.
Industry	Includes manufacturing, mining, agricultural facilities; extraction and disposal complexes; etc.
Health / Medical	Includes hospitals, research institutions, aid posts
Physiography	Description of terrain
Oceanography	Includes environments of the oceans
Vegetation	Includes natural and cultivation plant life
Climatology	Climatic phenomena

Figure 3 World view categories

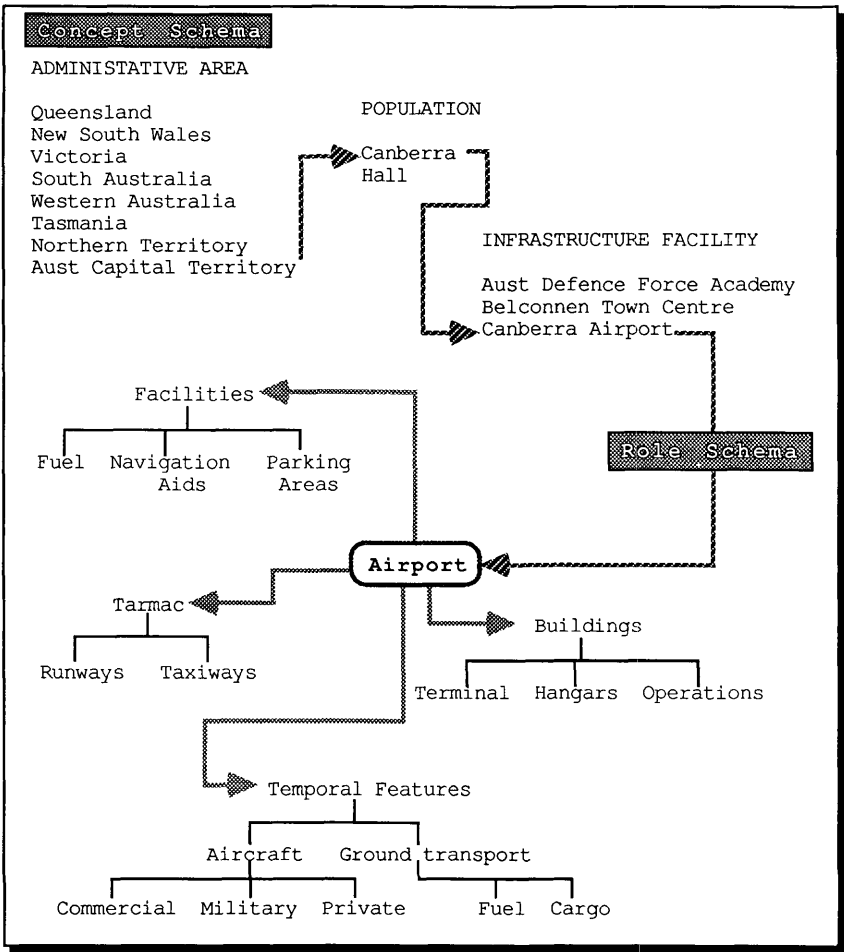


Figure 4 Concept and role schemas

Society is becoming information conscious. Each year there is an ever growing need to know more about the environment. Therefore there is pressure on those involved with land studies to provide accurate information and creditable advice on subjects pertaining to the environment. It seems apparent that the tools required to perform analyses on the *phenomena of the real world*, will come from the *expert systems* area of artificial intelligence but operating on geographic data structured according to *world view* concepts.

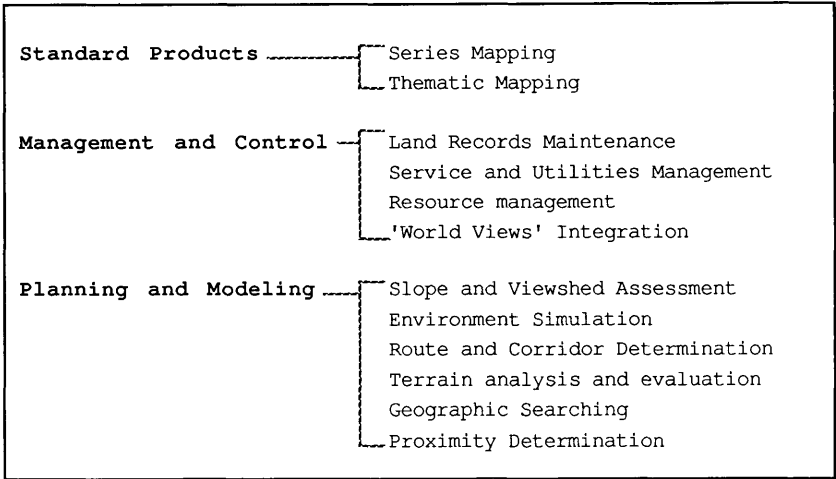


Figure 5 Some cognitive views of geographic information

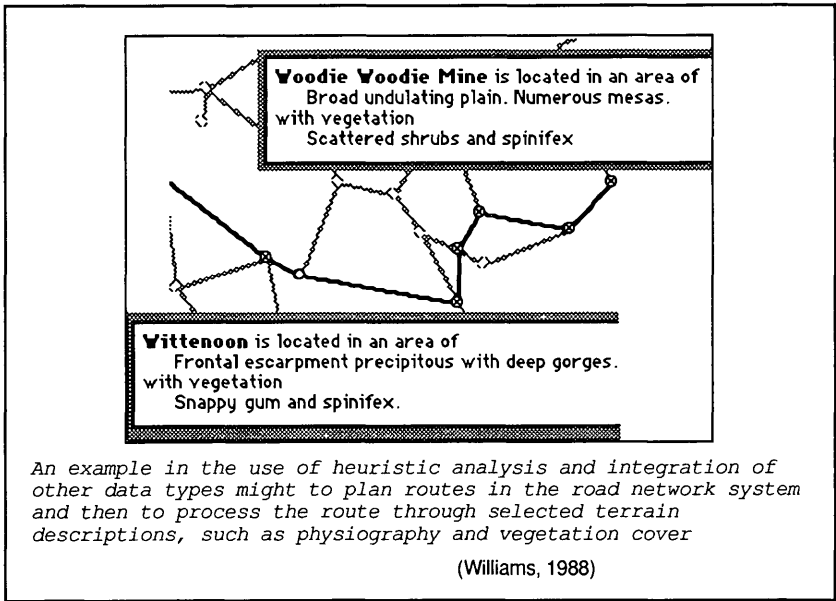


Figure 6 Analysis of the road transportation network

REFERENCES

- Bouillé, F. (1986). "Interfacing cartographic knowledge structures and robotics", **Proceedings - Auto Carto London**, Vol 2, Ed Michael Blakemore, International Cartographic Association.
- Grady, R.K. (1986). "New-age cartography", **Computer Graphics World**, 9, 10.
- Guptill, Stephen C., Kenneth J.Boyko, Michael A.Domaratz, Robin G.Fegeas and David E.Hair (1987). "An Enhanced Digital Line Graph Design", Internal Paper of the National Mapping Division, United States Geological Survey, Washington, DC
- Hayes-Roth, F, D.A.Waterman and D.B.Lenat (1983). **Building Expert Systems**, Addison-Wesley Publishing Company, Reading, Massachusetts.
- Moellering, Harold (Ed) (1987). "Issues in Digital Cartographic Data Standards", Report #8, National Committee for Digital Cartographic Data Standards, The Ohio State University, Columbus, Oh.
- Moore, G.T and R.G.Golledge (eds) (1976). **Environmental Knowing: Theories, Research, and Methods**, Dowden, Hutchinson and Ross, Stroudsburg,Pa.
- McKeown, David M. Jr (1983). "Concept Maps", Technical Report CMU-CS-83-117, Carnegie-Mellon University, Pittsburgh, Pa.
- Rhind, David (1988). "A GIS Research Agenda", **International Journal of Geographic Information Systems**, Ed. J.T.Coppock and K.E.Anderson, Vol.2, No.1,Taylor and Francis, London.
- Robinson, Arthur, Randall Sale and Joel Morrison (1978). **Elements of Cartography**, Ed. 4, John Wiley and Sons, New York.
- Rossmeissl, Hedy, Michael Domaratz, John List, Lynn Usery (1987). "Proposed Definition of Cartographic Features for Digital Line Graph-Enhanced (DLG-E)", Report of Committee Investigating Cartographic Entities, Definitions and Standards, National Mapping Division, USGS, Washington, DC.
- Trewartha, Glenn T., Arthur H. Robinson and Edwin H.Hammond (1967). **Elements of Geography**, Ed 5, McGraw-Hill, New York.
- Williams, R.J. (1987). "Evolution in cartography: data intelligence", **Cartography**, Vol 16, No 2, The Australian Institute of Cartographers.
- Williams,R.J.(1988) "Analysis of the Road Transportation Network", Paper presented to **7th Australian Cartographic Conference**, Sydney

GIS SUPPORT FOR MICRO-MACRO SPATIAL MODELING

Timothy L. Nyerges
Assistant Professor
Department of Geography
Smith Hall DP-10
University of Washington

ABSTRACT

Concepts, techniques and tools in geographic information systems (GIS) have come a long way primarily because of a) database management support for spatial data bases, b) spatial operators for point, line and area features - the current mainstay in spatial analysis as related to GIS, and c) graphics display. For years spatial modelers in human geography have been developing concepts and techniques for information analysis, however these developments have seldom found their way into a relationship with GIS. This paper indirectly explores reasons why this relationship has not developed faster to support model-based spatial analysis, given the length of time spatial modeling and GISs have been developing by examining the interrelationships between GIS and transportation modeling. Central to this problem has been the intended depth and narrow focus of spatial modeling applications versus the shallow breadth of GIS information management and analysis. GIS issues are presented in terms of system functionality. Spatial modeling issues are presented in terms of 'basic dimensions' in the context of spatial transportation modeling. The relationship between the two topics is explored by identifying GIS functions that can support various 'basic dimensions of modeling' in a transportation modeling context.

INTRODUCTION

Is it true that "(g)eographical information systems and model-based locational analysis: (are) ships (passing) in the night or the beginnings of a relationship?". (Birkin, Clarke, Clarke, and Wilson 1987 p 1). Birkin and others raise that question concerning the relationship between geographic information systems (GIS) and locational analysis in geography, but it could be considered for all socio-economic oriented modeling in geography including transportation modeling as well. They argue that "GISs and models in geography have largely grown through different cultures and traditions. Both schools have achieved obvious successes, but both have the same kinds of difficulty when it comes to practical application " (Birkin, Clarke, Clarke and Wilson 1987,1).

Birkin and others (1987) suggest that the history of GIS is tied rather closely to remote sensing (and by implication environmental sciences), and therefore GIS is inadequate when it comes to supporting social science studies. In addition, the authors claim that model

building in academic geography has suffered from a lack of practical context, and therefore argue for a more "applicable human geography" (Clarke and Wilson 1987). The same case has recently been made in the *Annals* in an article titled "Geography Beyond the Ivory Tower" (Demko 1988). Calling for a better integration of theory and practice is by no means new, having been made over the years and in every academic field. But what is different now in geography is that GISs are being adopted rapidly by practitioners in all segments of society. Perhaps conceptual advances with GIS in support of social science studies can help further advances in model building in human geography.

The goal in both GIS and model building environments has been the production of "information" from raw data, with the definition of "information" being its "usefulness" in a decision-making process. How this is done is a matter of the traditions and methodologies of the different schools.

Nyerges and Dueker (1988) discuss the spatial analysis capabilities in GISs oriented to transportation applications in terms of spatial operators and model-based software for either socio-economic or environmental problems. Spatial operators are currently the mainstay of the GIS spatial analysis due to the fact that most modeling environments are applications specific. Model-based software modules have tended to be rather large in many instances and therefore not easily absorbed into GIS environments or the expertise to perform such interfacing has not been available. Only a few implementations of socio-economic models in a software environment with some GIS-like capabilities have been reported (Babin, Florian, James-Lefebure and Spiess 1982; de la Barra, Perez and Vera 1984; Birkin, Clarke, Clarke and Wilson 1987). These latter implementations point out the need for a better understanding of how GIS functionality can support socio-economic, model-based analysis.

The relationship between GIS and socio-economic, model-based analysis in geography needs to be developed further in terms other than just an occasional interfacing of software, if the two topics are to experience mutual benefits over a long term. The fundamental basis of integration/interfacing is in need of further exploration.

This paper indirectly explores reasons why the relationship between GIS and socio-economic modeling has not developed faster to support model-based spatial analysis, given the length of time spatial modeling and GISs have been developing. Central to this problem has been the intended depth and narrow focus of spatial modeling applications versus the shallow breadth of GIS information management and analysis. However, we can look upon this problem as an opportunity to better understand the foci of both GIS and spatial modeling if we contrast and compare the fundamental characteristics of each. That is, the functionality in a generic GIS and the 'basic dimensions of spatial modeling' in human geography. The context of this research is an exploration of GIS support

for micro-macro spatial transportation modeling. In particular, travel demand forecasting models for state and urban transportation planning are being integrated/interfaced with a GIS environment.

GIS FUNCTIONALITY AND SPATIAL TRANSPORTATION MODELING

Transportation GIS Functionality

Several definitions for GIS have been proposed over the years, with perhaps as many definitions as there are view points on the subject. Cowen (1988) summarizes many of those definitions and identifies four basic approaches to defining a GIS. These approaches are: (a) a process-oriented approach in terms of input, storage, retrieval, analysis, and output, (b) an application which categorizes a GIS according to the type of data, (c) a toolbox approach incorporating a sophisticated set of computer-based procedures and algorithms for handling spatial data, and (d) a database approach as a refinement of the toolbox approach and focusing on the data retrieval performance of the system rather than the functionality. Cowen (1988, 1554) suggests that none of those approaches are suitable for a definition and concludes that "a GIS is best defined as a decision support system involving the integration of spatially referenced data in a problem solving environment." The focus is on decision making and problem solving. We can help clarify this GIS definition by turning to Bonczek, Holsapple and Whinston (1981, 3) who define a decision support system as "an information processing system that is embedded within a decision making system" whereby the decision making system might be human, machine or human-machine.

The definition of a GIS used here is most like that of the decision support orientation as favored in Cowen (1988). However, the modeling capability is left to functions that provide 'modeling support' rather than a specific 'modeling application'. The intent here is to identify functions in a GIS which can support modeling, and leave the system implementation as whether models are interfaced or integrated up to the system specific environment. A list of potential functions that might exist in a transportation oriented GIS appear in Table 1. All of these functions in some way support a modeling environment, but the interest here is in those that most directly support such an environment. The next section presents a list of 'basic dimensions of spatial modeling' to outline the issues that one considers when building a spatial-oriented socio-economic model, and in particular transportation models.

Table 1. Functionality in a Transportation GIS

Human interface

- Non-programming interface: an interface to support retrieval and analysis for both casual users such as menus and expert users such as a command language.
- Programmer's interface: a subroutine library interface to support programmer use for extending and creating applications.

Data capture and editing

- Digitizing: manual or automatic digitizing.
- Data validation: integrity constraints for data quality checks such as detection of 'overshoot' digitizing errors.
- Import/Export: ability to load in (import) or send out (export) bulk spatial and attribute digital data.
- Interactive editing: capability for a user to add/delete objects/data values one at a time with the use of retrieval criteria at the option of the user.
- Batch editing: capability for a user to add/delete objects/data values in bulk processing.
- Map edge match: matching the edges of maps by selecting a center line having a band width.

Spatial and thematic data management

- Map area storage/retrieval: continuous geographic domain for any area or group of areas can be stored and retrieved as a single database.
- Spatial data description: construction of point features, link/node topology with shape records, chain encoding of polygon boundaries.
- Locational reference: use of absolute referencing such as latitude/longitude, state plane or UTM coordinate reference system or relative referencing such as route-milepoint, address or other specialized system depending on problem orientation.
- Global topology: global network topology for any geographic domain and any set of data categories to be defined by system administrator at the request of users.
- Thematic data description: construction of attribute fields to qualitatively and quantitatively describe data categories.
- Data definition: software to manage descriptions and definitions of the data categories and the spatial and thematic data descriptors of these categories.
- Spatial selective retrieval: retrieval of data based on spatial criteria such as coordinate window, route-milepoint reference.
- Thematic selective retrieval: retrieval of data based on thematic criteria such as name of road, attribute of road.
- Browse facility: retrieval of any and all data categories.
- Access and Security: multi-user or single user access with read/write protection.
- Roll-back facility: supports restoration of database state in the event of system failure.
- Minimal data redundancy.
- Subschema capability: select parts of a corporate-wide database for special management.

- Database size: No limitation on the number of points, lines or areas per map, maps per data base, or coordinates per line or area should exist for logical storage of elements within the capacity of physical storage.

Data manipulation

- Structure conversion: conversion of vector to raster, quadrees to vector.
- Object conversion: point, line, area, cell, or attribute conversion to point, line, area, cell, or attribute.
- Coordinate conversion: map registration, 'rubber sheet' transformations, translation, rotation, scaling, map projection change or image warping.
- Locational classification: grouping of data values to summarize the location of an object such as calculations of area centroids, proximal features, Thiessen polygons.
- Thematic classification: grouping of thematic data values into classes.
- Locational simplification: coordinate thinning of lines.
- Locational aggregation: grouping of spatial objects into a superordinate object.
- Class generalization: grouping data categories into the same class based on characteristics of those categories.
- Thematic aggregation: creation of a superordinate object based on thematic characteristics of two or more other objects.

Data analysis

- Spatial object measurement: individual object and interobject calculations for line length, area and volume, distance and direction.
- Statistical analysis: frequency analysis, measures of dispersions, measures of central tendency (mean, median, mode), correlation, regression.
- Spatial operators: point, line, area object on/in point, line, area object; gravity model primitives; network indices for beta, diameter and accessibility.
- Routing: identify routes based on spatial or thematic criteria.
- Model structuring: a model structuring environment that provides linkages between parts of models perhaps through a special language.

Data display

- Symbolization change: any graphic symbolization could be created at the option of the user.
- Softcopy graphics: viewing of maps, graphs on CRT monitor.
- Hardcopy graphics: maps, graphs on printer/plotter.
- Reports: reports on content of database, report formatting to support analysis, formatted summary tables.
- Display window: the area of the database currently being examined.
- Overview window: a window used for quick spatial orientation that shows the entire geographic domain.
- Pan: the ability to roam across a geographic domain bringing data to the CRT screen without having to change the display window.
- Zoom: changing the area of the display window to examine more or fewer features, resulting in a change of scale of

the display image. A change in accuracy usually does not occur.

Note: The list of functions is a combination of a list appearing in Table 1 of (Rhind and Green 1988) and a list appearing in a bid specification created with the assistance of the author (State of Alaska 1986). The list as prepared by Rhind and Green (1988) is a synthesis of several authors' works. Several of the classes as appear in (Rhind and Green 1988) have been reorganized and further elaborated based on the authors experience in writing system specifications.

Dimensions of Spatial Transportation Modeling

Models in human geography as related to transportation systems have been a topic of interest for many years resulting in too many models to mention in detail. Good summaries of the breadth of the field appear elsewhere such as (Chorley and Haggett 1967, Wilson and Bennett 1985). Even these extensive presentations do not completely describe the effort as suggested in the review of (Wilson and Bennett 1985) as provided by MacKinnon (1987).

Focus in this research involves transportation modeling which is part of an urban transportation planning process. Meyer and Miller (1984) describe the urban transportation planning process as consisting of the following decision-oriented steps:

1. Diagnosis and data management
2. Analysis and evaluation
3. Scheduling and budgeting
4. Monitoring

Since the mainstay of GIS is data management and graphics it is natural that part of current and future GIS research be involved with data management and other functionality suitable to support analysis for certain application areas, thereby systematically linking the above mentioned Steps 1 and 2. Transportation analysis and evaluation is one of the most complex and lengthy portions of the transportation planning process. A majority of the analysis process involves transportation modeling. According to Werner (1985) the spatial transportation modeling process is made up of the following steps:

1. **Estimate trip generation characteristics.** Inflows and outflows from each traffic analysis zone (TAZ) must be estimated at an appropriate disaggregated scale using trip generation models.
2. **Compute trip distributions.** Characterize the origins and destinations of the TAZs and balance these to conserve the total number of trips across the transportation area using trip distribution models such as growth factor, intervening opportunities or gravity models.
3. **Estimate the modal split.** Determine the proportions of the different modes of transportation that apply to each of the origins and destinations using modal choice models.

4. **Assign trips to routes.** All trips must be assigned to routes to load the network using network loading models.

5. **Evaluate alternative network structure and use.** Alternative network designs and the operation of these networks according to vehicle access are examined using models based on dynamic programming (shortest path) and linear programming (optimal use) algorithms.

A systematic approach to describing the idiosyncrasies of spatial transportation models can start with a set of basic dimensions of any human-geography related modeling task as identified by Wilson (1987, 414): entititation, scale, spatial representation, partial versus comprehensive, static analysis and dynamic analysis. These basic dimensions identify the issues and considerations that need to be addressed when developing (or in this case interfacing or integrating) models to a GIS. Consequently, the models in the above steps could be described according to the six dimensions; each description being used to identify GIS functionality to support those models.

The six dimensions that characterize the basic considerations for building spatial-oriented models in human geography are describe in more detail as follows:

1. **Entititation** - Enumerate the basic components of the system of interest, e.g. what are the important entity types (categories of information) of a transportation system (Wilson 1981) such as transportation routes (rivers and freeways), routes, vehicles (multimodal - cars, buses, trains etc.).

2. **Scale** - What level of resolution should be adopted, e.g. should all entity types be populated with data to a level such that all vehicles and all roadways are considered in the analysis?

3. **Spatial representation** - Determine how to treat space. Treat space in a continuous way, so that exact locations of activities can be determined as in the case with every origin and destination of a trip. Treat space in a discreet way such that zonal boundaries are more appropriate as the spatial aggregations or origins and destinations.

4. **Partial versus comprehensive** - Are the location of any marginal activity considered in the analysis, for example a firm or a household, given the rest of the system as an environment or are all activities considered simultaneously, thus reflecting competitive processes?

5. **Static analysis** - What theory and methods are used for the static analysis of structure and form.

6. **Dynamic analysis** - What theory and methods are used for the dynamic analysis of evolution and change.

Wilson (1987) offers several observations clarifying these dimensions. The first four dimensions are of a different character than the last two. The first four determine the way we look at the system of interest for analytical purposes; the last two relate to detailed specification of the theories and methods.

Differences in creating a model can arise because of dimensional choices for data representation even though the theoretical basis may be the same. For example, continuous space economic models and discrete space economic models look different because the constructs used to create the model are different, even though their theoretical underpinnings are still the same. Continuous space economic models are much more likely than discrete zone models to necessitate restrictive assumptions such as "all employment at the city center" in order to make the mathematics practical. These differences result in the need for various functional requirements that match a particular modeling approach.

Wilson (1987) recognizes the importance for conditional forecasting that will remain useful for short-run planning, but emphasizes the need for a shift in perspective from long-run comprehensive, impact analyses which has not been very productive (Lee 1973) to analyses of stability and resilience of components in urban systems. Despite the differences in short-run partial versus long-run comprehensive approaches to modeling, Meyer and Miller (1984, 182) identify several issues that are common to both the long-run and short-run modeling tasks:

1. the dynamic nature of an urban area
2. the complexity of urban behavior
3. the need for good quality, detailed data

Providing support for the integration of micro-macro spatial transportation models that can address short-run and long-run issues at local and regional scales in a convenient and systematic way will test the capabilities of current and future GISs. In essence, a micro-macro approach is an attempt to build structured, comprehensive models while taking into consideration data availability constraints in the short-run, addressing these constraints and associated data requirements with GIS support.

Birkin and Clarke (1986) discuss micro-macro approaches to dynamic urban modeling. An example of a micro approach is a micro-simulation model of urban travel behavior. Such a model captures the decision making process of an urban traveler throughout the daily cycle of activities over the urban landscape. A macro approach uses trip data aggregated to origin and destination places represented as different traffic analysis zones (TAZs). The macro approach uses an accounting method for balancing all of the trips generated from origin zones to destination zones with the return trips from the destination zones back to the origin zones (Werner 1985).

GIS SUPPORT FOR TRANSPORTATION MODELING

No single GIS specification is likely to address all geographical applications, as no single GIS specification is likely to address all modeling applications, including transportation modeling. However, as a beginning to a systematic exploration of what GIS functions best support a modeling environment, the basic set of GIS functions outlined in Section 2.1 can be compared against the basic dimensions of modeling as described in the context of transportation modeling in Section 2.2. Each GIS function can be examined in turn as potentially providing some support to a transportation modeling environment, or alternatively each dimension can be considered as to how each function might support that dimension. The results are shown in Table 2.

Table 2. GIS Functionality for Transportation Modeling

GIS FUNCTIONALITY	MODELING DIMENSIONS					
	1	2	3	4	5	6
Human interface						
Non-programming interface	x	x	x	x		
Programmer's interface	x	x	x	x		
Spatial and thematic data management						
Map area storage/retrieval				x		
Spatial data description			x			
Locational reference			x			
Global topology				x		
Thematic data description	x					
Data definition	x					
Spatial selective retrieval			x			
Thematic selective retrieval	x					
Browse facility				x		
Access and Security				x		
Roll-back facility						
Minimal data redundancy				x		
Subschema capability				x		
Database size				x		
Data manipulations						
Structure conversion	x		x			
Object conversion		x				
Coordinate conversion			x			
Locational classification		x	x			
Thematic classification	x	x				
Locational simplification		x	x			
Locational aggregation		x	x			
Class generalization	x					
Thematic aggregation	x					
Data analysis						
Spatial object measurement			x			
Statistical analysis				x		
Spatial operators			x	x		
Routing				x		
Model structuring	x		x	x		

Some functionality is more directly related to the modeling interface/integration than is other functionality. The human interface, data management, data manipulation and data analysis functionality categories are directly related to a modeling environment. The data capture and data display functionality categories indirectly support integration/interfacing and are not considered in this evaluation (although this is the subject of related research).

For brevity, all models mentioned in Section 2.2 are taken as a general category, therefore only a single column exists as it applies to all models. A more detailed analysis would include a separate column for each model on each dimension. In addition, the dimensions for static analysis and dynamic analysis are not considered in this evaluation, since very little (if any) GIS functionality can truly support dynamic analysis to an extent of other than static time slices. Currently "time" in most data bases is represented as a thematic attribute. Such an approach is in need of further conceptual development (Langran and Chrisman 1988).

The following generalizations can be made with regard to human interface, data management, manipulation and analysis functionality for support of transportation modeling. Since entitation is a topic that involves the basic substance of what phenomena are being modeled, the GIS functionality that involves thematic attribute management - either naming them or deriving their descriptions from other data - is fundamental to developing the basic set of entities to be used in the modeling process. The spatial data management of locational data is fundamental to the spatial representation dimension, but from a static analysis point of view. That is, data management is concerned with the storing and retrieving of static descriptions. Current GISs would have difficulty with the dynamic representation of entity classes and/or their spatial representations.

Data manipulation functionality involves the conversion of one data object into another data object to support a particular scale of analysis. The scale of analysis dictates an appropriate data resolution for the data classes (determined through entitation). Data resolution indicates how general or detailed the data values are for the sampled entities. Different levels of detail can be supported through forward and reverse generalization, classification, aggregation and simplification processes of the data objects. This data manipulation supports changes in scale for a micro-macro approach to modeling.

The micro-macro approach involves more than a matter of spatial scale, but also is concerned with what can be called thematic scale and temporal scale. Map scale (as a term that is defined more precisely than spatial scale) is a term that refers to the relationship between distance on the ground and distance on a map. Hence spatial scale refers to how much distance, or what area is included,

focusing on spatial position. The term scale includes measurements for both resolution and accuracy. That is, the units of resolution for knowing how much distance or area is involved together with whether the measurement can be replicated with confidence is important. Thematic scale involves the degree of specificity for any given attribute and whether the measurement is accurate. The same would apply to time, as in the case of "when something occurred". The temporal characteristics of geographical information is a topic of current research (Langran and Chrisman 1988) in a GIS context.

Data analysis functionality involves basic spatial operators that are application specific, basic statistical analyses and spatial object measurement. The analysis functionality would primarily support a partial versus comprehensive approach to building models. Spatial object measurements could be made for the raw data to be submitted for statistical analyses. Statistical analyses for central tendencies and regression could support parameter estimation for the primitives of models. The spatial operators might include gravity model operators that are the primitives used to construct more sophisticated spatial interaction models. Spatial operators might also include network indices, such as beta diameter and accessibility indices, for describing the complexity of a network, hence being used to decompose the network into simpler terms for quicker analysis at a general level, then more detailed analysis at a local level.

SUMMARY and CONCLUSIONS

This paper started out by asking the question: "After years of development for both spatial modeling and GIS, are these developments passing as ships in the night or is there a relationship beginning to emerge?" If we assume the latter to be occurring; it will only pass a superficial level by taking a more systematic cross-fertilization of the two areas of expertise.

GIS solutions for many geographical problems are now being investigated. One of these is in transportation applications. Transportation modeling is beginning to come out of an experimental realm and is being put to use in several municipalities across the United States.

Several issues about GIS functionality to support spatial transportation modeling are still unresolved. The evaluation presented here is preliminary, and would most likely differ from one modeling problem to another. However, an overall indication of the nature of GIS and modeling functionality can be assessed using this approach.

Further refinement is needed of the approach used in identifying GIS functionality that can support transportation modeling. Rather than indicating "direct" or "indirect" support or a "yes" or "no" level of support, the level of support could be indicated along a scale and

prioritized for certain applications. This can be done through a more thorough examination of particular transportation models. This topic is currently under investigation through an examination of the use of TIGER/Line files as a base network for transportation modeling in the State of Washington.

REFERENCES

- Babin, A., M. Florian, L. James-Lefebure and H. Spiess 1982. EMME/2: Interactive graphic method for road and transit planning, *Transportation Research Record*, 866, 1-9.
- Birkin, M. and M. Clarke 1986. Comprehensive dynamic urban models: integrating macro- and micro- approaches, Working Paper 439, School of Geography, University of Leeds.
- Birkin, M., G. P. Clarke, M. Clarke, and A. G. Wilson 1987. Geographical information systems and model-based locational analysis: ships in the night or the beginnings of a relationship?, Working Paper 498, School of Geography, University of Leeds.
- Bonczek, R. H., C. W. Holsapple and A. B. Whinston 1981. *Foundations of Decision Support Systems*, New York : Academic Press.
- Chorley, R. J. and P. Haggett 1967. *Models in Geography*, London : Methuen.
- Clarke, M. and A. G. Wilson 1987. Towards an applicable human geography: some developments and observations, *Environment and Planning A*, 19:1525-1541.
- Cowen, D. J. 1988. GIS versus CAD versus DBMS: What are the Differences?, *Photogrammetric Engineering and Remote Sensing*, LIV(11):1551-1555.
- de la Barra, T. B. Perez and N. Vera 1984. TRANUS-J: putting large models into small computers, *Environment and Planning B*, 11(1):87-102.
- Demko, G. J. 1988. Geography beyond the ivory tower, *Annals of the Association of American Geographers*, 78(4):575-579.
- Langran, G. and N. R. Chrisman 1988. A Framework for Temporal Geographic Information, *Cartographica*, 25(4): in press.
- Lee, D. A. 1973. Requiem for large-scale models, *Journal of the American Institute of Planners*, 39(3):163-178.
- MacKinnon, R. D. 1987. Review of *Mathematical methods in human geography and planning*, by A. G. Wilson and R. J. Bennett 1985.
- Meyer, M. D. and E. J. Miller 1984. *Urban Transportation Planning*, New York : McGraw-Hill.

Nyerges, T. L. and K. J. Dueker 1988. Geographic information systems in transportation, Technical Report, U. S. Dept. of Transportation, Federal Highway Administration Planning Division, also included in Proceedings of the Computer-assisted Cartography in Transportation Conference.

Rhind, D. W. and P. A. Green 1988. Design of a geographical information system for a heterogeneous scientific community, *International Journal of Geographical Information Systems*, 2(2):171-189..

State of Alaska 1986. Bid Specification for a Digital Mapping System for the State of Alaska Department of Transportation and Public Facilities. Department of Administration, Bid No. 11451, Juneau, Alaska.

Werner, C. 1985. *Spatial Transportation Modeling*. Beverly Hills, CA: SAGE Publications.

Wilson, A. G. 1981. *Geography and the Environment: Systems Analytical Methods*. Chichester: John Wiley & Sons.

Wilson, A. G. 1987. Transport, location and spatial systems: planning with spatial interaction models, Chapter 7 in *Urban Systems: Contemporary Approaches to Modeling* (eds.) C. S. Bertuglia, G. Leonardi, S. Occelli, G. A. Rabino, R. Tadei, and A. G. Wilson, London: Croom Helm.

Wilson, A. G. and R. J. Bennett 1985. *Mathematical Methods in Human Geography and Planning*. Chichester, Sussex : John Wiley & Sons.

CONTEXT-FREE RECURSIVE-DESCENT PARSING OF LOCATION-DESCRIPTIVE TEXT

Matthew McGranaghan

University of Hawaii
Department of Geography
2424 Maile Way
Honolulu, HI 96822

ABSTRACT

Databases in which locations are specified in (near) natural language text, rather than as coordinates or topological relations pose difficulties for current GIS and automated mapping systems. Such databases may include metes and bounds descriptions of properties, or textual descriptions of locations where biological specimens were collected. The difficulty of converting textual location descriptions to coordinate data was highlighted by recent efforts to map the collection sites of specimens in the Bishop Museum's Herbarium Pacificum.

Human interpretation of both the text-based locational information in the Herbarium records and a number of topographic maps was required to derive mappable coordinates from textual descriptions. An automated system which could interpret the textual descriptions, and return coordinates, is an attractive alternative.

This paper reports an effort to create such a system by modifying a context-free recursive-descent text parser. A model of the grammar used in location descriptions is presented. The model recognizes phrases which denote specific features, generic features and terms relating them. The parser will recognize words as elements of these phrases. Meaning is ascribed to the location descriptions by relating them to items in standard geographic data sets (USGS DEMs, DLGs and GNIS).

INTRODUCTION

McGranaghan and Wester (1988) reported deriving geographic coordinates from textual descriptions of sites where herbarium specimens were collected. The process was slow and tedious. Because current GIS technology relies on analytic geometry and cartesian coordinate systems, and because much existing spatial data is not referenced to such coordinate systems, it seems likely that many scientists will be faced with a similar task in trying to use GIS technology.

To make the problem more concrete, each of the 80,000 specimens in the Bishop herbarium is associated with a unique label. This label contains a museum accession number, the plant's identification, the name of its collector, the method used to preserve the specimen, and a description of where the specimen was collected, but it does not contain systematic spatial coordinates for the site. The locational information includes the name of the island from which the plant was collected, a locality (the name of a physical feature or land division where it was collected), the approximate elevation at which it was collected, and (usually) a more detailed narrative description of the collection site.

The narrative descriptions are constrained by several practical concerns. They tend to be terse, composed by field scientists, and able to fit on a few lines of a label form. They also tend to be nearly procedural, giving directions that one could take to reach the same site again. Information about the site which is relevant to plant habitat, such as ground cover, soil type and moisture, are often included in the description. Virtually none of the descriptions are written as complete, grammatically correct, sentences. All of the information from the labels has been entered (verbatim) into a database as part of an effort to automate herbarium management.

For the initial mapping project, interpreting these descriptions was the slowest part of the data conversion process (McGranaghan and Wester 1988). It required map reading skill, and judgement. The amount of detail provided in these descriptions varies. Consequently, the confidence in the derived map locations varied.

Converting the descriptions to mappable coordinates involved sorting the descriptions, interpreting them, plotting these points and then digitizing them. The location descriptions were sorted by island and locality, then the narratives were used to plot the sites on the correct USGS 7.5 minute series topographic maps. This was done manually, and usually involved visually scanning a number of possible maps for the name of an area or feature. It was possible that the name did not appear on any of the maps. Once the area had been found, the rest of the text was interpreted to fix the point with respect to, for instance, topography, elevation, ground cover, and what ever other information the collector had provided. All of the plotted points on each map sheet were digitized at once, and the table coordinates converted to latitude and longitude.

This paper describes an approach to automating this conversion process. The goal of this research is a computer program which can read text describing a location and produce the absolute position of the site where a specimen was collected and an estimate of the confidence associated with the position. To "understand" the textual description, the program must be able to parse the text, identify phrases and terms which locate the position with respect to the planet, as represented in standard USGS data sets.

NATURAL LANGUAGE UNDERSTANDING

Making computers understand natural language has occupied computer scientists for several decades. During this time, some general strategies have developed, much has been learned about the complexity of the task, and several programs capable of understanding simple English sentences about fairly restricted domains have been produced (Winston 1977).

Strategies for understanding natural language attempt to exploit regularities and constraints found in the language to break a sentence into meaningful units. This process is called parsing. The constraints which allow one to parse a sentence are related to both the meanings of words in the sentence, and to sentence structure.

Both the words and the structure contribute to the meaning of a natural language sentence. Some of the words guide in determining the sentence structure, while others identify what referents which are

being related in the sentence. The sentence structure indicates how the things referred to by the words are related in the "meaning" of the sentence, and may guide expectations about where to find specific parts of the meaning. Sentence structure, or grammar, provides a great deal of information about the meaning of a sentence through the context it provides.

GRAMMAR AND PARSING

The Handbook of Artificial Intelligence (Barr and Feigenbaum 1981) provides an accessible introduction to formal languages and grammar. A grammar is a scheme for putting words together into the phrases and sentences allowed in a language. Grammars are generally defined as a tuple of elements and possible relations among them. Symbolically, a grammar can be represented as:

$$G(P,W,R,S)$$

Where the grammar (G) provides rules (R) relating a basic sentence (S) to a set of non-terminal phrases (P), which in turn are composed of members of a set of terminal units, words (W), available in the language. The intersection of P and W is the null set.

The rules are often represented as productions, in which the constituent parts of a non-terminal unit are indicated. The form of the rules can be used to classify the grammar. If the rules are such that a single non-terminal symbol is on the left-hand side of each production, the grammar is context-free. An example of such a grammar, drawn from The Handbook of Artificial Intelligence is:

```
<SENTENCE> -> <NOUN PHRASE> <VERB PHRASE>
<NOUN PHRASE> -> <DETERMINER> <NOUN>
<NOUN PHRASE> -> <NOUN>
<VERB PHRASE> -> <VERB> <NOUN PHRASE>
<DETERMINER> -> the
<NOUN> -> boys
<NOUN> -> apples
<VERB> -> eat
```

This grammar could generate sentences such as: "boys eat apples", "the boys eat apples" or "the apples eat the boys". "Eat the apples" would not be a valid sentence in this grammar because there is no <NOUN PHRASE> preceding the <VERB PHRASE> (that is, this statement is only a <VERB PHRASE> and not a complete <SENTENCE>).

Algorithms exist for parsing sentences which can be characterized by such a grammar. NLP.C (Schildt 1987) is a simple parser written in C which can process slightly more complex sentences. Its rules are:

```
<SENTENCE> -> <NOUN PHRASE> <VERB PHRASE>
<NOUN PHRASE> -> <NOUN>
<NOUN PHRASE> -> <DETERMINER> <NOUN>
<NOUN PHRASE> -> <DETERMINER> <ADJECTIVE> <NOUN>
<NOUN PHRASE> -> <PREPOSITION> <NOUN PHRASE>
<VERB PHRASE> -> <VERB> <NOUN PHRASE>
<VERB PHRASE> -> <VERB> <ADVERB> <NOUN PHRASE>
<VERB PHRASE> -> <VERB> <ADVERB>
<VERB PHRASE> -> <VERB>
```



```

<VERB> -> { list }
<NOUN> -> { list }
<DETERMINER> -> { list }
<ADVERB> -> { list }
<ADJECTIVE> -> { list }
<PREPOSITION> -> { list }

```

If the location descriptions followed these rules, NLP.C could parse them. By adding words to the data base in Schildt's NLP.C, it will parse, "the plant is in the valley." into a <NOUN PHRASE>, "the plant" and a <VERB PHRASE>, "is in the valley."

The NLP.C parser uses a recursive algorithm to parse an input sentence. Its routines find the parts of speech of words in the sentence, and use them to determine how to parse the sentence. The routines are mutually recursive, and the order in which they are called indicates the phrase structure of the sentence.

To determine the end of the <NOUN PHRASE> and the beginning of the <VERB PHRASE> NLP.C simply takes the first <NOUN> it encounters to be the end of the <NOUN PHRASE>. Similarly, the <VERB PHRASE> begins with a <VERB>, though it may end several ways.

PARSING LOCATION DESCRIPTIONS

The following examples of location description narratives are drawn from the herbarium database:

Niu

wet forest

Kaulani, on open hillside

In woods near base of pali directly back of Kaimi Farm,
Koolau Mts.

Ko'olau Mts., along the Waikane-Schofield Trail

Ko'olau Range, Waikane-Schofield Trail, in woods along
trail

South ridge of Kipapa Gulch, Waipio

Higher gulches

North Fork of Kipapa Gulch. Koolau Mts. Along stream and
up the banks at elevation of 1100-1500 ft.

2nd Gulch E. of Pu'u Kaupakuhale, N.E. slope of Pu'u
Ka'ala, moist bottom of gulch

Ridge North of Waimea Valley

The formal grammar in NLP.C does not adequately characterize these descriptions. The location descriptions are not proper sentences. Most do not contain a <VERB PHRASE>, and many contain several <NOUN PHRASE>s. In short, the plant location parser must deal with a somewhat less structured "sentence" than does NLP.C.

Still, there is some structure. The descriptions are composed of one or more "location-descriptive phrases". There is some regularity in the structure of these phrases. The order in which they appear is less regular than NLP.C would expect, and the phrase-types often are repeated in these "sentences". Inducing a grammar for the location descriptions requires identifying the forms of the structures used.

As an aside, the structure of these descriptions may reflect some feature of human spatial cognition. A location may be defined by intersecting constraints, to distinguish a location from all others. The organization of the constraints may not be important; rather, the meaning comes from the combination of them. There may be some advantage to listing the constraints from most general to most specific (closely related to procedural directions for finding the site). Details like "moist bottom of gulch" may only be useful if the location has already been limited to a particular gulch. However, even that pattern is not always used.

A model of this grammar must allow a description to be composed of one or more location description phrases. The location description phrases have a number of forms. They tend to be composed of nouns and modifiers. The nouns name either generic features (stream, ridge, gulch, etc.) or specific features such as Waikane-Schofield Trail, Kipapa Stream, or Kaimi Farm. The modifiers are usually prepositional phrases. A model of the grammar used in the descriptions might be represented as:

```

<LOCATION DESCRIPTION> -> <LOC DES PHRASE>*
  <LOC DES PHRASE> -> <NOUN PHRASE> | <NOUN> |
    <PREPOSITIONAL PHRASE>
  <PREPOSITIONAL PHRASE> -> <PREPOSITION> <NOUN PHRASE>
    <NOUN PHRASE> -> <DETERMINER> <NOUN> | <ADJECTIVE> <NOUN>
      | <DETERMINER> <ADJECTIVE> <NOUN>
        <NOUN> -> <SPECIFIC FEATURE> | <GENERIC FEATURE>
  <SPECIFIC FEATURE> -> <UNKNOWN> <GENERIC FEATURE> | { gnis } |
    <UNKNOWN>
    <DETERMINER> -> { list }
    <PREPOSITION> -> { list }
    <ADJECTIVE> -> { list }
    <GENERIC FEATURE> -> { list }

```

In this grammar, determiners, prepositions, and adjectives are considered closed-classes; each set contains a fairly small and fixed number of terms. As Leonard Talmy pointed out last June in Buffalo (Talmy 1988), such terms mark the grammatical structure of language. A parser can use them to track the structure of a sentence and, in turn, to identify the words that should refer to geographic features.

Another closed-class is being posited in the current version of the grammar. This is the "generic feature". Generic features are common landscape elements, such as "hill", "valley" or "stream". Two distinct sources for the members of this set were identified. The first was an exhaustive examination of the words in the herbarium database. The list produced this way included names for features of great local significance, such as "pali". Another, more standard, source of generic features is the set of "Feature Class Definitions" used in the USGS Geographic Names Information System (GNIS). This set

of 63 terms for generic landscape features has the advantage of being documented and identical for the whole country. Terms with considerable local usage could be either added to this list or translated to the most appropriate term in the list.

Parser Strategy

In the NLP.C parser, the parts of speech of the words encountered by the parser signal which grammatical phrases the words belong to. Given the nature of the locational phrases, it seems that parsing these location descriptions amounts to recognizing sets of prepositional phrases, and interpretation will then be determining which data bases contain the nouns. An example of how the parser can recognize the parts of a prepositional phrase follows:

<PREPOSITIONAL PHRASE> -> <PREPOSITION> <NOUN PHRASE>

The <PREPOSITION> is the clue that a <PREPOSITIONAL PHRASE> is beginning. The function in the parser which recognizes <PREPOSITIONS> indicates that one has been encountered. A second <PREPOSITION>, when encountered, marks the close of this <PREPOSITIONAL PHRASE> and the beginning of another. Within a <PREPOSITIONAL PHRASE> there must be one or more words, which must be identified as parts of a <NOUN PHRASE>. These in turn, must be decomposed into some combination of determiners, adjectives, generic features and specific features. When the description has been broken down into its constituent parts, and each part's functions determined, the parsed description still needs to be interpreted.

Interpretation Strategy

Parsing the text is only part of the job. To assign meaning to the text, it must be interpreted. The information gained from parsing, will be used in the context provided by standard geographic databases (USGS Digital Elevation Models, GNIS, Digital Line Graphs and US SCS soil facet maps), to deduce the coordinates. Remember that in addition to the data derived through parsing, the Herbarium labels also provided a locality name and elevation data to use as a starting point in determining a location.

The parsed description produces a set of descriptive phrases. Each of these phrases can be thought of as a constraint on the described location. The role of each word in the description is known (or inferred) from its position a in phrase.

The prepositions indicate the spatial relations among the features identified in the descriptions. Containment and enclosure are indicated by "in" and "on". Position with respect to linear features might be indicated by "along". Proximity may be indicated by "near", "by" and others.

When a specific feature name can be found in the GNIS, coordinate determination becomes a "look-up" operation. Preliminary testing indicates that a high proportion of specific features named in the descriptions will be found in the GNIS database. This is a result of both the GNIS database and the collector's descriptions being derived from USGS topographic maps. The positions found this way may be modified by other parts of the description.

The generic features do not give a direct look-up key to geographic data sets but they may provide information to guide geographic pattern-matching search in several data sets. GNIS records generic feature names associated with mapped items. Even when a specific feature name is not available in the description, or does not match in GNIS, it is possible that the generic feature can be used to produce a list of possible sites of the right type. Together with other constraints this may prove sufficient to derive a location's coordinates.

Generic feature names might also indicate topographic configurations which might be recognizable in DEM or DLG data. For instance, a "hill" or a "draw" might be recognized as a particular configuration of elevations in a DEM. This type of pattern matching may not be exceedingly expensive if the rest of a description sufficiently limits the region to search. See O'Callaghan and Mark (1984), Band (1986) and Frank, Palmer and Robinson (1986) for discussions of techniques which might be employed and problems which must be overcome in this type of matching. Search in a DEM is further constrained by elevation data from the label data.

Adjectives and non-feature nouns also contain information that might be useful if other data sets are available. These might be especially useful if information about soil types, land use/land cover, and climate are available and spatially referenced.

The interpretation engine will need to resolve the set of constraints produced by the parser to a single location or set of possible locations. This involves spatial reasoning about the relations indicated by the description in light of information found in the standardized data sets. This is clearly a step beyond the parser.

FUTURE DIRECTIONS

The most pressing need in this project is to refine and generalize the grammar understood by the parser. In addition to improved utility of the parser, it is expected that this will aid understanding of how people conceive of, and describe, locations. Further evaluation of the value of conceptualizing specific and generic features as separate classes, and of considering generic features to be a closed set is needed.

A second objective is to make the parser more robust. One complication with joining diverse data sets is the need to match place names given spelling variations. The USGS data sets do not use diacritical marks in place or feature names. Considerable pride in the Hawaiian language, and a desire to maintain it, have resulted in many field scientists in Hawaii retaining the use of macrons, apostrophes (for glottal stops), and other diacritical marks. Diacritical marks are inconsistently used on the Herbarium labels and pose problems for word-matching software.

In the longer term, another goal is to develop a spatial reasoning system which can use knowledge from a wide range of domains, as does a human interpreter, in determining locations. Knowledge, such as the habitat normally associated with a species, or the time-space history of the collector, or even personal habits of collectors could be used.

REFERENCES

- Band, L., 1986, "Topographic Partition of Watersheds with Digital Elevation Models", Water Resources Research, v. 22, n. 1, pp. 15-24.
- Barr, A., and Feigenbaum, E. A., eds., 1981, The Handbook of Artificial Intelligence, vol. 1, William Kaufmann, Inc.
- Frank, A., Palmer, B., and Robinson, V., 1986, "Formal Methods the for Accurate Definition of Some Fundamental Terms in Physical Geography", Proceedings: Second International Symposium on Spatial data Handling, July 5-10, Seattle, Washington, pp. 583-599.
- McGranaghan, M., and Wester, L., 1988, "Prototyping an Herbarium Collection Mapping System", Technical Papers: 1988 ACSM-ASPRS Annual Convention: GIS, v.5, pp. 232-238.
- O'Callaghan, J., Mark, D., 1986, "The Extraction of Drainage Networks from Digital Elevation data", Computer Vision, Graphics and Image Processing, v. 28, pp. 323-344.
- Schildt, H., 1987, "Natural-Language Processing in C", Byte, December 1987, pp. 269-276.
- Talmy, L., 1988, Presentation at a two day workshop, "Cognitive and Linguistic Aspects of Space", June 11-12, 1988, State University of New York at Buffalo.
- Winston, P. H., 1977, Artificial Intelligence, Addison-Wesley.

Object-Oriented Modeling in GIS: Inheritance and Propagation*

Max J. Egenhofer
Andrew U. Frank

National Center for Geographic Information and Analysis
and

Department of Surveying Engineering
University of Maine

Orono, ME 04469, USA
MAX@MECAN1.bitnet
FRANK@MECAN1.bitnet

Abstract

The relational data model has proven to be too restrictive for applications with spatial data, such as Geographic Information Systems (GIS). In particular, the absence of techniques to form complex objects and represent spatial objects at different abstraction levels makes it difficult to model geographic situations properly. The object-oriented approach, which has been recently promoted for similar engineering applications, such as CAD/CAM, VLSI design, or molecular models in chemistry, seems to overcome some of the deficiencies. By incorporating the abstraction mechanisms *generalization* and *aggregation*, the data model gets richer and more powerful than the relational model, and the application designer is given more and better tools to model complex situations.

Both generalization and aggregation involve the derivation of attribute values at different levels of detail and abstraction. Two methods for the derivation of properties are introduced: (1) *inheritance* describing properties and methods of subclasses in is-a hierarchies, and (2) *propagation* deriving properties in part-of hierarchies. While inheritance acts in a top-down fashion along the generalization hierarchy, propagation can derive values from parts to the aggregates (bottom-up). Frequently aggregate functions, such as SUM, MIN, or MAX are involved to pass on properties of composed objects to its parts.

1 Introduction

The relational model upon which most current GIS software systems are built has been acknowledged as an insufficient model for applications that deal with spatial data [Frank 1984] [Härder 1985] [Frank 1988b]. The application of object-oriented techniques for the design of future Geographic Information Systems has been proposed on several stages, e.g., such as message-passing programming language [Kjerne 1986], object-oriented database management systems [Egenhofer 1987], and object-oriented software engineering techniques [Egenhofer 1989a] [Egenhofer 1989b]. This paper focuses on advanced object-oriented techniques to model the dependencies of properties, operations, and values in generalization- and aggregation hierarchies. These methods are ideal for applications with spatial data, because they enforce natural phenomena, such as the fact that the area of a subdivision is exactly the sum of the areas of the partitions.

*This research was partially funded by grants from NSF under No. IST 86-09123 and Digital Equipment Corporation. The support from NSF for the NCGIA under grant number SES 88-10917 is gratefully acknowledged.

Barrera and Buchmann introduced the derivation of attributes in hierarchies of spatial inclusion and aggregates to geographic applications [Barrera 1981]. Since then, many controverse discussions have been among researchers in the areas of *object-oriented modeling*, *object-oriented database management systems* [Dittrich 1986b] [Dittrich 1988], and *object-oriented programming languages* [OOPSLA 1986a] [OOPSLA 1986b] about various types of inheritance, such as behavior vs. abstract implementation, single vs. multiple, automatic vs. on-demand, and upwards vs. downwards.

The paper begins with an overview of the principles of object-oriented modeling. Examples of GIS applications show the benefits of the abstraction mechanisms classification, generalization, association, and aggregation, for spatial modeling. There are two methods for relating properties from one object to another. In section 3 the data model is extended by *inheritance* linking properties of objects in a generalization hierarchy. In section 4 *propagation* is introduced linking the values of objects that are linked by association or aggregation. The separation of these two different methods is important, therefore, clearly distinct terms are chosen. Propagation is sometimes called in the literature *upward inheritance* [Barrera 1981] [Brodie 1984a], but it should become clear from this paper that there exist two different concepts which must not be mixed. The paper concludes that inheritance and propagation are important for GIS applications and need efficient support from object-oriented programming languages.

2 Object-Oriented Model

This chapter introduces the notation of objects and the abstraction tools available to deal with them. A definition of object-orientation is that an entity of whatever complexity and structure can be represented by exactly one object [Dittrich 1986a] No artificial decomposition into simpler parts should be necessary due to technical restrictions, e.g., normalization rules [Codd 1972]. The object-oriented data model is built on the four basic concepts of abstraction [Brodie 1984b]: classification, generalization, association, and aggregation.

2.1 Classification

Classification is the mapping of several objects (instances) to a common class. The word *object* is used for a single occurrence (instantiation) of data describing something that has some individuality and some observable behavior. The terms *object type*, *sort*, *type*, *abstract data type*, or *module* refer to types of objects, depending on the context. In the object-oriented approach, every object is an instance of a class. A type characterizes the behavior of its instances by describing the operators that are the only means to manipulate those objects [O'Brien 1986]. All objects that belong to the same class are described by the same properties and have the same operations. Classification is often referred to as the *instance of* relationship because the individuals are *instances of* the corresponding class

For example, the GIS model for a town may include the classes *residence*, *commercial building*, and *street*. A single instance, such as the residence with the address '30 Grove Street' is an instance of the class *residence*. Operations and properties are assigned to object types, so for instance the class *residence* may have the properties *number of bedrooms* and *address* which are specific for all residences

2.2 Generalization

Generalization groups several classes of objects, which have some properties and operations in common, to a more general superclass [Dahl 1966] [Goldberg 1983]. The terms *subclass* and *superclass* characterize generalization and refer to object types which are related by an *is.a* relation. The converse relation of superclass, the *subclass*, describes a specialization of the superclass. For example, the object type *residence* is a *building*; *residence* is a subclass of

building, while *building* is its superclass¹.

Generalization may have an arbitrary number of levels in which a subclass has the role of a superclass for another, more specific class. The *building/residence*-generalization can be extended with the class *rural residence*. While *residence* is a subclass of *building*, it is at the same time a superclass for *rural residences*.

It is important to note that superclass and subclass are abstractions for the same object, and do not describe two different objects. The residence with the address '30 Grove Street', for example, is at the same time an instance of the class *residence* as well as of its superclass *building*.

2.3 Association

Association is a form of abstraction in which a relationship between similar objects is considered a higher level set object [Brodie 1984b]. The term *set* is used to describe the association, and the associated objects are called *members*. Hence, this abstraction is referred to as the *member_of* relation, but is also often called *grouping* or *partitioning*. For example, a subdivision divides one parcel into several parcels.

The details of a member object are suppressed and properties of the set object are emphasized. An instance of a set object can be decomposed into a set of instances of the member object. Association applied to objects (members) produces a set data structure. An operation over a set consists of one operation repeated for each member of the set, e.g., a FOR EACH loop structure, found in some modern programming languages, such as CLU [Liskov 1981].

For example, the *city* Orono and the *building* with the address '30 Grove Street' are associated by the relationship *inside*.

2.4 Aggregation

A similar abstraction mechanism to association is aggregation which models composed objects, i.e., objects which consist of several other objects [Smith 1977]. The term *composit* object describes the higher-level object, while *subpart* or *component* refers to the parts of the composit object. The relationship among the components and the composit object is the *part_of* relationship, and the converse relationship is *consists_of*. For example, the class *building* is an aggregate of all *walls*, *windows*, *doors*, and *roofs* which are *part of* it.

When considering the aggregate, details of the constituent objects are suppressed. Every instance of an aggregate object can be decomposed into instances of the component objects. Each part keeps its own functionality. Operations of aggregates are not compatible with operations on parts.

Aggregation applied to objects (components) produces an aggregate (or record) type data structure. An operation over an aggregate consists of a fixed number of different operations in sequence or in parallel, one for each component. Hence, aggregation relates to sequence or parallel control structures.

3 Inheritance

In generalization hierarchies, the properties and methods of the subclasses depend upon the structure and properties of the superclass(es). Inheritance is a tool to define a class in terms of one or more other, more general classes [Dahl 1966]. Properties which are common for superclass and subclasses are defined only once—with the superclass—and inherited by all objects of the subclass, but subclasses can have additional, specific properties and operations which are not *shared by the superclass*. *Inheritance* is the transitive transmission of the properties from one

¹Frequently, the terms *parent* and *child* are used for superclass and subclass, respectively. Though this terminology is helpful to clarify the dependency of subclasses from superclasses, it is not accurate with respect to the abstraction, because the relationship between parent and child is not *is_a*.

superclass to all related subclasses, and to their subclasses, etc. This concept is very powerful, because it reduces information redundancy [Woelk 1987] and maintains integrity. Modularity and consistency are supported since essential properties of an object are defined once and are inherited in all relationships in which it takes part.

Operations of the superclass are applicable to all objects of the subclass because each object of the subclass is at the same time an object of the superclass; however, operations which are specifically defined for a subclass are not compatible with superclass objects.

3.1 Single Inheritance

The inheritance relation can be restricted to form a strict hierarchy and is then often referred to as *single inheritance*. *Single inheritance* requires that each class has at most a single immediate superclass. This restriction implies that each subclass belongs only to a single hierarchy group and one class cannot be part of several distinct hierarchies.

Figure 1 shows an example of inheritance along a generalization hierarchy. *Residence* is the general superclass and *city residence* and *rural residence* are the specific subclasses. All properties of the class *residence* are inherited to the two subclasses. For example, *residentName* is a property of the class *residence* which applies to all *city residences* and *rural residences*, and hence is inherited to them. Likewise, all operations defined upon *residence*, such as *moving into a residence*, are applicable both to *city residences* and *rural residences*. On the other hand, the operations defined specifically for a subclass are not applicable for objects of the superclasses. For example, *nextSubwayStop* is a property which applies only to city residences.

The common representation of hierarchies as trees is used for strict inheritance with the most general superclass at the top, and the most specific subclasses at the bottom. Each class is modelled as a node, while the *is_a* relation between two nodes is visualized as a vector pointing from the node of the superclass to the node of the subclass. The direction of the vector is to emphasize the top-down concept of inheritance—from the general to the specific.

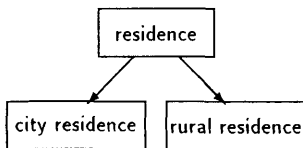


Figure 1: An example of inheritance along a generalization hierarchy with the more general class at the top and more specialized classes at the bottom

The transitive property of inheritance implies that any property is passed not only from the superclass to the immediate subclasses, but also to their sub-subclasses, etc. Figure 2 shows a more complex generalization hierarchy with 3 levels of classes. The properties of a *building*, such as *address* and *owner*, are inherited to the subclass *residence*, and also transitively to the sub-subclasses *rural residence* and *city residence*.

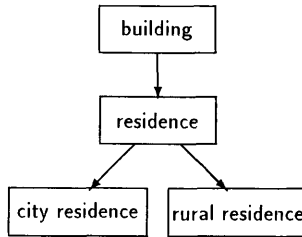


Figure 2. Properties are transitively inherited from a superclass to all its subclasses, the sub-subclasses, etc

3.2 Formalism

The inheritance relation can be described formally in First Order Predicate Calculus. First Order Predicate Calculus is a language based upon a set of primitive symbols composed of (1) variables, constants, and predicate symbols; (2) logical connectors NOT, AND, OR, implication, and equivalence; (3) quantifiers FOR ALL and EXISTS; and (4) parentheses. A combination of constants and variables (called predicates), linked with the logical connectors, is called a well-formed formula (*wff*)

Subsequently, constants (or facts) are capitalized, while variable names are lower cased. Facts and rules (axioms) will be marked by an asterisk (*) to distinguish from inquiries about predicates (hypothesis). Each property of a class is expressed as a predicate of the form *p (class, property)*. Generalization is described as the *is_a*-predicate of the form *is_a (subclass, superclass)*. The following facts describe the model depicted in figure 2.

```

*P (Building, Address).
*P (Building, Owner).
*P (Residence, Resident).

*is_a (RuralResidence, Residence).
*is_a (CityResidence, Residence).
*is_a (Residence, Building).
  
```

Inheritance is then defined by a predicate *properties* which recursively derives the properties associated with a class and all its superclasses.

```

*properties (class, property) IF p (class, property).
*properties (class, property) IF is_a (class, superclass),
    properties (superclass, property).
  
```

All properties of the class *cityResidence* can then be determined with the predicate

```

properties (CityResidence, prop).
  
```

which is fulfilled with the following values for the variable *prop*:

```

prop = Resident
prop = Address
prop = Owner
  
```

Likewise, it can be determined to which classes a certain property belongs, e.g.,

```

property (class, Resident).
  
```

yields

```

class = Residence
class = RuralResidence
class = CityResidence

```

This example showed the need for the definition of properties only once, that is with the most general superclass. All dependent properties are derived with the transitivity rule.

3.3 Multiple Inheritance

The structure of a strict hierarchy is an idealized model and fails frequently when applied to real world data. Most 'hierarchies' have at least a few non-hierarchical exceptions in which one subclass has more than a single, direct superclass. Very often more than one hierarchy of classes exists which is used concurrently. Again, the one-superclass-per-subclass rule is violated. Thus, pure hierarchies are not always the adequate structure for inheritance. Instead, the concept of *multiple inheritance* [Cardelli 1984] permits to pass operations or properties from several higher-level classes to another class. This structure is not hierarchical, because—in terms of the parent-child relation—one child can have several parents. Figure 3 shows the simplest case of multiple inheritance with a subclass inheriting properties from two distinct superclasses.

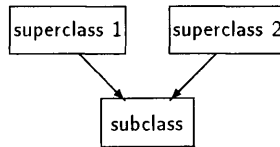


Figure 3: Multiple Inheritance: The properties of 2 distinct superclasses are passed to a common subclass

An example from geography shows how multiple inheritance combines often two distinct hierarchies. One hierarchy is determined by the separation of *artificial* and *natural* transportation links, whereas the other hierarchy distinguishes *water bodies*. Classes with properties from both hierarchies are *channels*, that are *artificial transportation links* and *water bodies*, and *navigable rivers*, that are *ivers* and *natural transportation links*. Other classes, such as *highway* or *pond* belong only to one hierarchy (Figure 4).

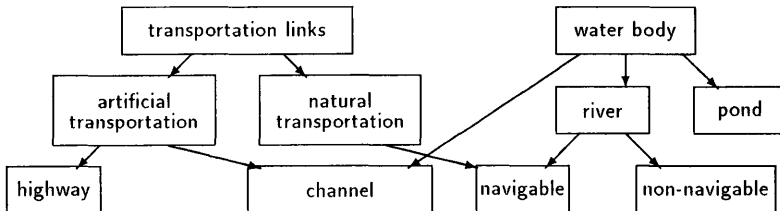


Figure 4: An example of multiple inheritance

3.4 Inheritance in GIS

Inheritance plays an important role for the clear and concise definition and implementation of very large software systems, such as Geographic Information Systems. The tools to implement such a system should be at least as powerful as the tools used for the conceptual model.

The advantages are similar to the ones of semantic models: Complex situations can be described concisely, consistency can be achieved by avoiding redundancy; and systems can be

maintained easier. A specific problem of the implementation of a GIS is the coexistence of a number of fairly complex tasks, such as the treatment of geometry, graphical representation, concurrent sharing of data, management of history and versions, etc.

Inheritance can be used as a software engineering design tool to describe the structure and properties of a GIS. One part of an application model is the definition of a set of classes as the abstraction of objects with common properties. Traditionally, for each class the appropriate operations and relationships must be defined, including operations which combine objects of different classes. For example, the class *building* has the operation *inside* which checks whether a building is located inside a parcel. Since *inside* applies also to many other objects, such as *cities* with respect to *counties*, many similar, often highly redundant operations are defined and implemented which make modifications difficult and yield frequently inconsistencies.

The application of inheritance overcomes these problems. By the definition of a general superclass for each specific concept, common properties may be defined in a single high-level class and inherited to the classes of the GIS application. Such a framework may consist of general superclasses, such as *spatial*, *graphical*, *temporal*, and *db-persistent*.

For example, the superclass *spatial* defines the geometric properties, such as location, spatial relationships, and spatial operators. A class in the user model can be defined as a subclass of *spatial* inheriting all its properties. For example, the class *building* is a spatial object. *Building* can be described as the subclass of *spatial* inheriting all spatial properties, such as the operation *inside* (Figure 5)

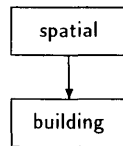


Figure 5. Creating a spatial class *building* by inheriting the spatial properties from a superclass *spatial*.

Other properties can be defined in a similar way. For example, database properties, such as persistency, multi-user access, and transaction control, can be inherited from a superclass *db-persistent*. The general database operations, such as store, delete, retrieve, and modify, are defined for the class *db-persistent* and passed to the specific object classes. If the class *building* is a db-persistent class, then buildings can be stored, deleted, retrieved, and modified.

It is obvious that this type of modeling requires multiple inheritance [Frank 1988a]. A class can have a multitude of diverse properties to be inherited. Important properties for GISs are *db-persistent* providing database behavior, *spatial* inheriting a common geometric concept, *graphical* providing graphical display, and *temporal* for the description of history of data [Egenhofer 1988]. Figure 6 shows the creation of the class *building* with two properties: *spatial* and *db-persistent*

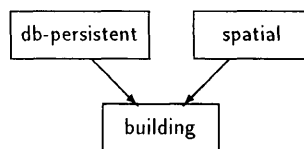


Figure 6. Creating a class *building*, the instances of which being both spatial and persistent.

4 Propagation

Aggregation and association show significant similarities, among them the ability to combine multiple instances to a combined instance. The dependencies of values along these hierarchies and their derivation is the subject of propagation. Subsequently, only aggregation hierarchies will be considered, but the same concepts apply to association hierarchies as well.

In aggregation hierarchies, two types of property values occur: (1) values which are specifically owned by the composite object and distinct from those of their components; (2) values which depend upon values of the properties of all components and must be derived from them. In contrast to less powerful models which require redundant storage of such values, the object-oriented model derives these dependencies. This model is superior because it enforces integrity by constraints. The population of a county, for example, is the sum of the population of all related settlements; therefore, the value for the property *population* of a *county* is derived by adding all values of the property *population* owned by the *settlements*.

4.1 Concept

The mechanisms to describe such dependencies and ways to derive values is called *propagation*. It supports complex objects which do not own independent data and is based upon the concept that values are stored only once, i.e., for the properties of the components, and then propagated to the properties of the composite objects. This model guarantees consistency, because the dependent values of the aggregate are derived and need not be updated every time the components are changed.

Propagation becomes trivial if the 'complex' object happens to be composed of a single part and the value of the aggregate refers to a single value of the part.

If the values of more than one object contribute to the derived value, the combination of the values must be described by an aggregation function. Aggregation functions combine the values of one or several properties of the components to a single value. This value reduces the amount of detail available for a complex object.

It may determine the sum or union of values of the components, or define a specific, outstanding part, such as the greatest, heaviest, or conversely, the smallest or lightest one. On the other hand, it may be representative, such as the average or weighted average of the values of a specific property.

Common operations for the derivation of values are *sum*, *set union*, *minimum*, *maximum*, *count*, *average*, and *weighted average*. For example, the population of the biggest city in a county is the maximum of the populations of all its cities; the area of a state is the sum of the areas of all its counties; the population density of the state is the average of the population density of its counties weighted by the county areas.

4.2 Propagation in GIS

In GIS, for example, a large number of attribute values at one level of abstraction depends upon values from another level. When combining *local* and *regional* data, this concept must be used to model the dependencies among data of different levels of resolution [Egenhofer 1986].

4.3 Formalism

The formalism of propagation can be described concisely in predicate calculus. The following simplified facts describe a county (Penobscot) as an aggregate of two settlements (Bangor, Orono) with the property *settlementPopulation*.

- *p (Orono, SettlementPopulation, 10,000).
- *p (Bangor, SettlementPopulation, 50,000).
- *p (Orono, PartOf, Penobscot).
- *p (Bangor, PartOf, Penobscot).

The population of the county is a value which is propagated from the values of the population of the associated settlements.

The population of a county is the sum of the population of the settlements. This dependency is expressed by the following rule, meaning the population of a specific county is the sum of the population of all settlements which are part of it.

```
*propagates (PartOf, SettlementPopulation, CountyPopulation, BySumming).
```

The generic rule for propagation is the following predicate. It describes the value of the property of an aggregate in terms of the values of the components using a specific aggregation function.

```
*p (aggregateClass, aggregateProperty, aggregateValue) IF
  propagates (relation, componentProperty, aggregateProperty, operation),
  p (componentClass, relation, aggregateClass),
  p (componentClass, componentProperty, componentValue),
  p (operation, componentValue, aggregateValue).
```

For example, the value of the property *countyPopulation* is then evaluated with

```
p (County, CountyPopulation, x).
```

and results in

```
x = 60,000
```

Propagation guarantees consistency because data is only stored once and derived from there. Updates underlie the common rules for updates of views [Dayal 1978], i.e., no derived properties can be updated explicitly, but only the fundamental properties. For example, it is not allowed to update the population of the Penobscot county by assigning the value 65,000 to the property *countyPopulation* if the town population of Orono grows by 5,000. Instead, the population of the settlements must be modified, e.g., `*p (Orono, SettlementPopulation, 15,000)`, which implicitly updates the *countyPopulation*.

5 Conclusion

The object-oriented model has powerful tools for data structuring, such as classification, generalization, aggregation, and association. In order to model dependencies of properties, operations, and values in hierarchies of generalized and aggregated objects, the concepts of inheritance and propagation are introduced. By using these techniques, complex situations in Geographic Information Systems can be modeled more naturally than with relational tables.

Three important conceptual differences exist between inheritance and propagation: (1) Inheritance is defined in generalization (*is.a*) hierarchies, while propagation acts in aggregation (*part_of*) or association (*member_of*) hierarchies. (2) Inheritance describes properties and operations, while propagation derives values of properties. (3) Inheritance is a top-down approach, inheriting from the more general to the more detailed class. Propagation on the other hand acts bottom-up.

Implementations need efficient support for these techniques. For example, programming languages must include object-oriented language constructs to model generalization and inheritance; to loop over aggregation parts; and to defined propagation.

6 Acknowledgement

Thanks to Renato Barrera and Alex Buchmann for many stimulating discussions which contributed to our understanding of object-orientation and propagation.

References

- [Barrera 1981] R Barrera and A Buchmann Schema Definition and Query Language for a Geographical Database System IEEE Transactions on Computer Architecture: Pattern Analysis and Image Database Management, 11, 1981.
- [Brodie 1984a] M.L Brodie and D Ridjanovic On the Design and Specification of Database Transactions In: M.L Brodie et al., editors, On Conceptual Modelling, Springer Verlag, New York, NY, 1984
- [Brodie 1984b] M.L. Brodie On the Development of Data Models. In: M.L. Brodie et al., editors, On Conceptual Modelling, Springer Verlag, New York, NY, 1984
- [Cardelli 1984] L Cardelli A Semantics of Multiple Inheritance. In G. Kahn et al., editors, Semantics of Data Types, Springer Verlag, New York, NY, 1984
- [Codd 1972] E.F. Codd Further Normalization of the Data Base Relational Model. In: R. Rustin, editor, Data Base Systems, Prentice-Hall, Englewood Cliffs, NJ, 1972.
- [Dahl 1966] O -J Dahl and K. Nygaard SIMULA—An Algol-based Simulation Language. Communications of the ACM, 9(9), September 1966
- [Dayal 1978] U Dayal and P Bernstein On the Updatibility of Relational Views. In: S. Bing Yao, editor, Fourth International Conference on Very Large Data Bases, West-Berlin, Germany, 1978
- [Dittrich 1986a] K Dittrich Object-Oriented Systems The Notation and The Issues. In: K. Dittrich and U Dayal, editors, International Workshop in Object-Oriented Database Systems, Pacific Grove, CA, 1986.
- [Dittrich 1986b] K Dittrich and U Dayal, editors Proceedings of the International Workshop in Object-Oriented Database Systems, Pacific Grove, CA. Springer-Verlag, New York, NY, 1986.
- [Dittrich 1988] K Dittrich, editor Advances in Object-Oriented Database Systems—Proceedings of the 2nd International Workshop on Object-Oriented Database Systems, Bad Münster am Stein-Ebernburg, F R Germany. Springer-Verlag, New York, NY, September 1988 Lecture Notes in Computer Science, Vol. 334.
- [Egenhofer 1986] M. Egenhofer and A Frank Connection between Local and Regional: Additional 'Intelligence' Needed. In: FIG XVIII. International Congress of Surveyors, Commission 3, Land Information Systems, Toronto, Ontario, Canada, 1986.
- [Egenhofer 1987] M. Egenhofer and A. Frank Object-Oriented Databases: Database Requirements for GIS In International Geographic Information Systems Symposium: The Research Agenda, Crystal City, VA, November 1987.
- [Egenhofer 1988] M Egenhofer Graphical Representation of Spatial Objects: An Object-Oriented View. Technical Report 83, Surveying Engineering Program, University of Maine, Orono, ME, July 1988.
- [Egenhofer 1989a] M. Egenhofer and A. Frank. PANDA: An Extensible DBMS Supporting Object-Oriented Software Techniques. In: Database Systems in Office, Engineering, and Scientific Environment, Springer-Verlag, New York, NY, March 1989.
- [Egenhofer 1989b] M. Egenhofer and A. Frank. Why Object-Oriented Software Engineering Techniques are Necessary for GIS. In: International Geographic Information Systems (IGIS) Symposium, Baltimore, MD, March 1989

- [Frank 1984] A. Frank. Requirements for Database Systems Suitable to Manage Large Spatial Databases. In: International Symposium on Spatial Data Handling, Zurich, Switzerland, August 1984.
- [Frank 1988a] A. Frank. Multiple Inheritance and Genericity for the Integration of a Database Management System in an Object-Oriented Approach. In: K.R. Dittrich, editor, Advances in Object-Oriented Database Systems—Proceedings of the 2nd International Workshop on Object-Oriented Database Systems, Bad Münster am Stein-Ebernburg, F.R. Germany, Springer-Verlag, New York, NY, September 1988. Lecture Notes in Computer Science, Vol. 334.
- [Frank 1988b] A. Frank. Requirements for a Database Management System for a GIS. Photogrammetric Engineering & Remote Sensing, 54(11), November 1988.
- [Goldberg 1983] A. Goldberg and D. Robson. Smalltalk-80. Addison-Wesley Publishing Company, 1983.
- [Härder 1985] T. Härder and A. Reuter. Architecture of Database Systems for Non-Standard Applications, (in German). In: A. Blaser and P. Pistor, editors, Database Systems in Office, Engineering, and Scientific Environment, Springer Verlag, New York, NY, March 1985. Lecture Notes in Computer Science, Vol. 94.
- [Kjerne 1986] D. Kjerne and K.J. Dueker. Modeling Cadastral Spatial Relationships Using an Object-Oriented Language. In: D. Marble, editor, Second International Symposium on Spatial Data Handling, Seattle, WA, 1986.
- [Liskov 1981] B. Liskov et al. CLU Reference. Lecture Notes in Computer Science, Springer Verlag, New York, NY, 1981.
- [O'Brien 1986] P. O'Brien et al. Persistent and Shared Objects in Trellis/Owl. In: K. Dittrich and U. Dayal, editors, International Workshop in Object-Oriented Database Systems, Pacific Grove, CA, 1986.
- [OOPSLA 1986a] OOPSLA '86—Object-Oriented Programming Systems, Languages, and Applications, Conference Proceedings. Portland, OR, September 1986.
- [OOPSLA 1986b] OOPSLA '87—Object-Oriented Programming Systems, Languages, and Applications, Conference Proceedings. Orlando, FL, October 1986.
- [Smith 1977] J.M. Smith and D.C.P. Smith. Database Abstractions: Aggregation. Communications of the ACM, 20(6), June 1977.
- [Woelk 1987] D. Woelk and W. Kim. Multimedia Information Management in an Object-Oriented Database System. In: P. Stocker and W. Kent, editors, 13th VLDB conference, Brighton, England, 1987.

GEOGRAPHIC LOGICAL DATABASE MODEL REQUIREMENTS

Martin Feuchtwanger

Department of Surveying Engineering, University of Calgary
2500 University Dr. NW, Calgary, Alberta, T2N 1N4, Canada

ABSTRACT

An important problem of GIS technology is the proper storage and retrieval of geographic data, and the logical database model, i.e. the approach taken in specifying the structure and meaning of and the operations performed on the stored data, is fundamental to its solution. A geographic logical database model should obey all the principles of:

a) syntactic database models, including file integration, controlled redundancy, unified data language, centralized access, independence, abstraction, concurrency, distribution, integrity and security; b) semantic database models, including abstract objects, relationship types, object-oriented query, knowledge incorporation, relativism and evolvability; and c) geographic data processing, including taxonomy, temporality, symbology, geometry, uncertainty, theme integration and view generalization. When such a model is provided, the design of geographic databases for sophisticated GISs will be possible.

INTRODUCTION

The purpose of this paper is to outline the requirements of a logical database model suitable for GIS.

A database is an organized set of interrelated data and is the central component of any information system. The design of the structure of a database is known as a schema and it is specified using a data language. The software package that defines the structure of and handles all access to a database is termed a database management system (DBMS).

What distinguishes different kinds of DBMS is an underlying theory known as a database model. It can be defined as a conceptual tool for describing data (or entity) types, their relationships, operations and constraints. Different classes of data models vary according to how closely they are oriented toward human or machine understanding, respectively. Logical database models, e.g. the entity-relationship, semantic, relational and network data models, help us specify what is going on. Physical database models relate to how the logical specifications are implemented and are not further discussed.

Consider the following: a geographic database (GDB) is a database appropriate for a GIS; a geographic DBMS (GDBMS) is a special DBMS controlling the nature and content of a GDB; and a geographic database model is a theory guiding the design of a GDBMS. What sort of logical database model do we want for GIS? One that will enable us to build, maintain and use sophisticated geographic databases. The model has a

number of important requirements. Some are specifically geographic and others apply to all information systems.

Although this paper is nominally practical, for the design of geographic databases, it could claim to move toward a general theory of geographic information.

MODEL TYPES and USERS and DATABASE VIEWS and DOMAINS

Four questions help guide the rest of the discussion.

What are the Main Types of Database Model?

Logical database models themselves can be subdivided according to their levels of abstraction.

The earlier ones are relatively low-level and are called datalogical (or syntactic) data models. They are also referred to as record-based because the entities involved are files, records, links and fields.

Newer database models are relatively high-level and are described as being infological (or semantic). They are also known as object-based because the entities involved are sets, objects, relationships and attributes.

Who Uses a Database Model?

The users of the database model can be considered to be database designers, application programmers or application programs, i.e. people or software, the distinction does not matter for this discussion. Strictly, they do not use the model directly, they use a DBMS, but again the distinction does not matter most of the time. End-user issues, such as "natural" query languages are not discussed.

What are the Major Views of a Geographic Database?

A view is a logical component of a database, as seen by a particular user. It is not necessarily all that exists.

The first major view of the database is that it is a real world simulation, an environmental model. It contains all of the data required for most purposes and may be called the phenomenal view because it describes the phenomena of interest.

The second major view of the database is that it is a symbolic (or visual) representation of the real world. It is derived from the phenomenal view by some design (or rendering) process and it may be called the cartographic view because it describes essentially a map.

Both views are digital and both exist at different levels of generalization (or scales).

What are the Major Domains of a Geographic Database?

Within each view (or portion) of a database are stored numerous kinds of data. Each datum (or piece of data) can

be considered to be drawn from a general domain, a class of data types, and each describes a different characteristic (or attribute) of an object or concept being represented in the database. Five major categories are identified.

Taxonomic (or thematic) data on an object tell us what it is. Examples include names, classes and values. They are often labeled simply as "attributes" although here the word refers to the more general concept.

Temporal (or historical) data, e.g. epochs and periods, on an object tell us when it is or was.

Symbolic (or visual) data, e.g. annotation, colors and shadings, on an object tell us what it looks like on maps.

Geometric (or spatial) data on an object tell us where and what shape it is. They can be either metrical or topological and include locations, coordinates and neighbors.

Scientific (or theoretical) data on an object tell us why it is, and may include laws, explanations and production rules.

Each domain is relevant to both the phenomenal and the cartographic views, except the symbolic domain which applies only to the cartographic view.

GENERAL REQUIREMENTS

The requirements of general-purpose database models are grouped according to how closely they relate to the world of the application environment relative to the workings of computer systems.

Characteristics of Standard Record-based Data Models

A geographic DBMS should at least exhibit all the characteristics of a conventional DBMS [Frank 1988]. Details can be found in most standard database texts. Briefly they are:

Integration. Several separate but related files are somehow combined into one unified whole, a database.

Controlled redundancy. The same data are not being duplicated, at least not unnecessarily or in such a way that inconsistency is possible.

Unified data language. A means of specifying both the structure of and the atomic operations to be performed on a database is provided.

Centralized access. Only one means of accessing the data, a common and controlled one, is used by all.

Independence. Programs are independent of the way data are stored, and data are independent of the way programs are implemented. Changes to one will not affect the logical characteristics of the other.

Abstraction. The same data exist at (or are viewed at) different levels of abstraction. Details of data viewed at one level are hidden from those at the next higher level.

Concurrency. Several users may access the same data at the same time.

Distribution. One large data set is stored or made available at several different sites.

Integrity. There is protection against illegal operations being performed on certain objects and against inappropriate objects being operated on by certain procedures.

Security. Data are protected against unauthorized access and against loss due to system failure.

The problem with such database models for any sophisticated information system is their limited semantic expressiveness.

SEMANTIC EXPRESSIVENESS of OBJECT-BASED DATA MODELS

A geographic DBMS must perform much more of the work that is required to properly manage GIS data than is currently the case. Much burden is placed on the application programs to perform many data management tasks, such as integrity maintenance. That leads to duplication of effort and program-data dependence, both contrary to the goals of database systems. Users of a GDB must have a facility for expressing concepts meaningful and useful to an application environment without excessive use of programs. They require data models with more semantic power.

Data semantics are the meaning, structural properties (i.e. objects and relationships), operational characteristics and integrity constraints of data [King & McLeod 1985; Su 1986]. Ideally, semantic data accurately reflect real-world objects or concepts. Following are some of the major requirements for achieving semantic expressiveness in database models.

Structures

The structure of a database, also called database statics, include various different object and relationship types.

Objects. A data model must provide a set of useful generic data types. As well as character, integer and real, for example, vector and tessellation could be provided. Also, it must be possible to combine the basic types into more complex ones, to suit particular applications. Ideally, there must be a simple, direct correspondence between an entity in the real world and an entity held in the database. Such is the notion of the abstract object [King & McLeod 1985]. Objects should be allowed to represent themselves directly instead of using some artificial identifier. Users should not have to be concerned with polygon IDs, for example, if such codes have no meaning in real life.

Relationships. The model must explicitly provide for different kinds of relationships (or associations) between entities [King & McLeod 1985; Su 1986]. Three are fundamental.

When an entity represents some action, mapping or relationship between two or more entities there is said to be an interaction type of relationship. For example, a land ownership represents a mapping between a land parcel and a land owner, and two roads may interact at an intersection.

When one or more entities together describe (or are attributes of) another usually more complex entity, there is an aggregation type of relationship. For example, a set of districts may form a region, and a name, a location and a population may describe a city.

When one or more entities are each subtypes of another more general entity there is said to be a generalization. For example, fir, spruce and pine are types of conifer, and transportation line may be a superclass for highway, railway and waterway.

Operations

The allowed operations (or transactions) on a database are also known as database dynamics. Basic database operations include retrieval, printing, deletion and insertion of an object based on certain attribute value conditions. Other ones involve whole sets of objects. Advanced operations might include certain display or mathematical functions on objects.

Expressing procedures as a series of basic operations is said to be navigational, in which case a high-level operation must be formulated using many query steps. Even with some non-navigational languages, where most operations can be specified with single queries, an intimate knowledge of the data structure is still required. In which case, a high-level operation must be formulated using a very complex query.

A user should not be burdened with using such a complex data language. A so-called object-oriented query mechanism should be provided whereby high-level transactions can be specified very simply.

Knowledge

If a GIS is going to be a decision-support system, produce well-designed maps, maintain a high degree of integrity, or live up to any more of its many expectations, it will have to be founded not just upon a database but upon a knowledge-base [Karimi & Feuchtwanger 1989].

A knowledge-base can be considered to be a database, extended to incorporate knowledge concepts. As well as the data (facts or assertions) that are in a database, it has expert rules for inferring new facts or rules from existing ones. Rule types include production rules, and semantic and security integrity constraints. They may be general or application specific. Analogous to the DBMS will be a knowledge-base management system (KBMS). It has an inference mechanism for applying the stored rules and an ability to explain the knowledge-base structure.

Thus, the data model must provide for the incorporation and use of knowledge into the schema by making the knowledge definable along with the objects to which it applies.

Relativism

Any GDB is likely to have several different classes of user, each one having different assumptions or expectations about the structure, operations and contents of the database. Each user may see the same object in a different way and have quite different authority regarding retrieval, update or analysis operations. Take the case of a lot. Different attributes of an entity "parcel" are of interest to different users --

surveyors: lot boundary measurements and owner;

planners: census data of its inhabitants;

engineers: location and specifications of its utilities.

Also, "owner" may be seen as just an attribute of an entity "lot" to one user but to another it is seen as an independent entity "person" joined by a relationship "owns."

One ought to be able to conceptualize different views of the same information as a semantic unit [King & McLeod 1985]. The model must have the ability to allow alternate views of and authority over a data set by different users. Such a concept is known as relativism.

Evolvability

A comprehensive GIS involves so many different data themes, accessed by many different users, at several sites that it is unlikely that any early database design is going to be adequate for all subsequent applications. Once a schema has been set and the database has been populated, it should be possible to modify the schema without the contents being corrupted. A model must possess evolvability, i.e. have the ability to allow a schema to evolve with changing knowledge or specifications of the application environment [King & McLeod 1985].

GEOGRAPHIC REQUIREMENTS

Geographic information systems are generally larger and more complex than most other types of information system. They have a number of significant logical database modelling requirements that are additional to those presented above. A set of particularly geographic objects, attributes, relationships, operations and constraints must be provided by the model. Space does not permit a full coverage of such constructs, so only a few examples appear below. The requirements discussed are those relating to the taxonomic, temporal, symbolic and geometric domains, presented above, plus those concerning data uncertainty, layer integration and levels of generalization.

Taxonomy

Conventional hierarchical classification schemes may be inadequate for complex GIS databases. Multi-branched tree structures with super object types, object types and sub-object types are called generalization hierarchies when represented using a logical data model. They can be too simple or restrictive, however, when an object is considered

to belong to more than one immediate class of objects. A river, for example, may simultaneously belong to the following classes: transport route, national boundary, drainage channel and waterbody. To model such situations a generalization network is required [Su et al. 1988]. More examples of generic objects that may be required in the taxonomic portion of the model include: thematic layer, categorical coverage, linear network, region and pointal feature.

Temporality

As the environment changes, so too must the database by which it is modelled, if the GIS is to maintain usefulness. However, a historical record is often useful too. As well as the spatial domain that commonly characterizes a GIS, there must be a temporal domain to the model. That is, the model must be capable of handling the monitoring of what changes take place, where and when they occur [Langran & Chrisman 1988]. It should contain data types such as date and period and should facilitate the answering of questions such as "what was the condition at location X in 1986?" and "when did the condition at location X change to situation S?" or certain kinds of time series analysis.

Symbology

Within a GDB, at any given scale, there must be a single view of the phenomena and possibly several corresponding views that are cartographic. The phenomenal view would be independent of graphic symbology. The cartographic views (or maps) contain their own symbology and would be derived from the phenomenal view by a map design (or rendering) process. That way, important data on any aspect of the phenomena and any kinds of analysis done on them are kept logically separate from the way they are visually represented. Any visualization of the phenomena is likely to contain only a partial subset of the phenomenal view and must involve a sophisticated cartographic design process, if it is to be optimal.

The model must therefore exhibit what may be called phenomena-cartography independence, the former being independent and the latter being dependent. Generic cartographic object type examples might include: line symbol, point symbol, label, legend and title. A simple cartographic operation type is "display an object" while a very high-level example would be "design a map."

Geometry

For any given geographic object (in either the phenomenal or the cartographic view) there may be more than one alternative geometric structure used to spatially represent it. Many have been proposed over the years and because different structures are optimal for different purposes more than one may be desired, even for the same object.

A single class of geometric objects may be used to represent many different kinds of geographic object, depending on the

application. For example, a polygon can represent a forest stand, a geological outcrop, a cadastral lot or a terrain patch. Also, several application objects may be simultaneously represented by the same instance of a geometric object. For example, several linear features may be spatially represented by the same polyline. However, the application level user must not be aware of (or be encumbered by) the particulars of the raster/vector implementation.

Thus, the model must support a level of abstraction at which geographic concepts are expressed independently of the geometry used to represent them [Feuchtwanger 1985]. The concept may be called application-geometry independence, the application level being independent of the geometric level. The latter contains many different spatial object, relationship, operation and constraint types.

Geometric object examples include: 0-cell (or point), 1-cell (or line), 2-cell (or polygon), 3-cell (or patch), tessellation (or image), run-length-coded bitmap, region quadtree, etc. Spatial relationships include: adjacency (or primary neighbor), proximity (or secondary neighbor), exhaustive subdivision, discontinuous homogeneous group, etc. Spatial retrieval operations include: the objects overlapping an object, the nearest object to a point, the subregions constituting a region, the geographic objects represented by a geometric object, etc. Geometrical integrity constraints include shape preservation and topological consistency [Mepham 1989; Zhang 1989].

Uncertainty

All taxonomic, temporal and spatial data are uncertain to some degree, whether from phenomenal fuzziness, measurement error or machine imprecision [Miller et al. 1989]. Categories of uncertainty include accuracy, precision and resolution. The uncertainty associated with (or quality of) all geographic data must be accounted for if the GIS is going to be credible. Also, the propagation of uncertainty during data processing and data combination must be handled. The model must have explicit facilities for associating uncertainty attributes to data and for appropriately combining uncertainties during data operations.

Integration

In a GIS there will be several different themes (or layers) of data relating to the same region. For many analysis applications, the integration (or overlay) of two or more of these layers will be required. Relationships between different layers can be explicitly represented within the database and retrieved when required, or derived computationally (either by the DBMS or by application software) when required. Since space is taken to store them, or time is taken to compute them, care is needed during design. For other applications, the different layers are otherwise quite unrelated and should be kept apart.

Thus, the model must allow for different degrees of

permanent integration to be specified. The concept could be called variable integration where the relationships between elements of different layers may be all, partly or not stored. An example of permanent integration is when the overlay of two polygon networks yields a composite network of common, smaller polygons.

Generalization

The ability to model the environment at different levels of generalization (or detail) is an essential characteristic of any sophisticated GIS. All views of reality are subjectively dependent on the scale of investigation and it is important that the model facilitates views of differing levels of generalization. Strictly, each level is dependent on its more detailed level and cannot be updated except via a special process, the generalization process.

For example, a small-scale route map might be derived from a large-scale topographic map. If a new road is built, the topography is updated and then appropriate data is propagated to the route map. A complete GIS may have a series of different phenomenal views, with only the most detailed one being updatable from the outside. The concept might be called multiple generalization and seems to contrast with the myth of the scale-free database. Generalization operations range from the low-level line simplification algorithms to the complex generalization of an entire thematic layer.

PRACTICAL CONSIDERATIONS

Although the model is a conceptual tool there are a number of practical things to consider when developing it.

Usable. It must be possible to use the model, i.e. to build suitable GIS schemas and to easily specify the storage and retrieval of geographic objects. It would be possible by means of the data language that must accompany the model.

Implementable. The model must not remain only theoretical; it must be implementable in the form of a GDBMS. That is the physical database design problem. How it is implemented raises many more questions that are beyond the scope of this paper. Briefly, the GDBMS may exist as an extension to some existing DBMS or as a completely redesigned package. Either way, special attention must be paid to efficient spatial access [Frank 1988].

Modifiable. The development of a logical database model for GIS is both a new and ongoing project. For the present, the model should be in a state of flux, not be fixed in stone; there must be room for modification, expansion or improvement.

General. From a computing point-of-view, the model must be more special-purpose than general-purpose, i.e. it facilitates geographic database design and use. From a geographic point-of-view, it must be general and simple

enough to be useful for most purposes, i.e. it is not so special or unduly complicated that it excludes certain applications. Finding a happy medium might be called the model designer's generality problem.

CONCLUSION

To conclude, a summary of the requirements of, implications of the existence of and recommendations for providing a geographic semantic database model are given.

Summary

A geographic logical database model should obey all the principles of: conventional database models, general semantic database models and geographic data processing.

Standard database principles include file integration, controlled redundancy, unified data language, centralized access, program-data independence, levels of abstraction, access concurrency, and data distribution, integrity and security. Semantic database principles include abstract objects, relationship types, object-oriented query, knowledge incorporation, relativism and evolvability.

Additional GIS principles involve: taxonomy (i.e. the generalization network), temporality, symbology (i.e. phenomena-cartography independence), geometry (i.e. application-geometry independence), uncertainty, integration (i.e. variable integration) and generalization (i.e. multiple generalization).

Implications

Current database models are inadequate even for general information systems. The relational model, for example, does describe structures and operations but has very limited semantics. Current semantic database models may also be inadequate for GIS. The entity-relationship model, one of the first semantic models, only describes database statics not dynamics. Neither kind expressly handles geometrically based applications.

A model exhibiting all of the specifications outlined above may be a long time in coming, but it will facilitate the design and use of sophisticated GDBs because it will have a much closer association with concepts in the world of geography.

Recommendations

Many of the above problems have been individually attacked by other researchers, but an integrated approach is needed. Also, much work is still to be done on developing the model. In particular, a comprehensive, integrated and formal set of geographic semantics should be produced and existing semantic data models should be investigated to see how well they cope with the added geographic requirements.

REFERENCES

- Feuchtwanger, Martin, 1985, An Investigation of Efficient Computer Techniques for the Storage and Retrieval of Land-related Information, M.Sc. Thesis, Department of Surveying Engineering, University of Calgary, Alberta, Pub. no. 20010.
- Frank, Andrew U., 1988, "Requirements for a Database Management System for a GIS," PE&RS, 54:11, 1557-1564.
- Karimi, H.A. and M. Feuchtwanger, 1989, "Geographic Knowledge Base Management System (GKBMS): The Future Challenge in Geomatics," presented paper, National GIS Conference, Ottawa, February/March.
- King, Roger and Dennis McLeod, 1985, "Semantic Data Models," in Principles of Database Design, Volume I, Logical Organizations, S. Bing Yao, ed, Englewood-Cliffs, New Jersey: Prentice-Hall, 115-150.
- Langran, Gail and Nicholas R. Chrisman, 1988, "A Framework for Temporal Geographic Information," unpub. paper, Department of Geography, University of Washington, Seattle.
- Mepham, Michael P., 1989, Automated Cadastral Data Entry Into an LIS, unpub. Ph.D. Thesis, Department of Surveying Engineering, University of Calgary, Alberta.
- Miller, R., H.A. Karimi and M. Feuchtwanger, 1989, "Uncertainty and its Management in Geographic Information Systems," presented paper, National GIS Conference, Ottawa, February/March.
- Su, Stanley Y.W., 1986, "Modeling Integrated Manufacturing Data with SAM*," IEEE Computer, 19:1, 34-49.
- Su, Stanley Y.W., Vishu Krishnamurthy and Herman Lam, 1988, "An Object-oriented Semantic Association Model (OSAM*)," in AI in Industrial Engineering and Manufacturing: Theoretical Issues and Applications, Kumara et al. eds, American Institute of Industrial Engineers.
- Zhang, Guangyu, 1989, Consistency Preserving Transactions for Automated Cadastral Database Systems, unpub. M.Sc. Thesis, Department of Surveying Engineering, University of Calgary, Alberta.

GIST: AN OBJECT-ORIENTED APPROACH TO A
GEOGRAPHICAL INFORMATION SYSTEM TUTOR

J.F.Raper and N.P.A. Green
Department of Geography, Birkbeck College,
7-15 Gresse Street, London W1P 1PA

ABSTRACT

Experience gained in the construction of the world's first GIS tutor, ARCDEMO (Green 1987), has emphasised the importance of accommodating different student learning strategies. ARCDEMO, while highly successful (it has been accessed over 2000 times via JANET-the UK Joint Academic Network) suffered from static graphical displays, a single predetermined access path, and an overall design which made alteration and updating of its material problematic. Object-oriented programming languages offered a means of addressing these problems and were also attractive in respect of the low development resources required. The Geographical Information System Tutor (GIST) exploits this approach using Apple's HyperCard software and incorporates a "point-and-click" interface with graphical cues to initiate operations which include animated demonstrations, step-by-step illustrative graphics and graphical displays capable of user modification. The topics covered by GIST, when taken together, define a set of core activities within GIS which can be used as the basis for a curriculum.

INTRODUCTION

The study of Geographical Information Systems (GIS) is a new and rapidly growing field in geography. These systems, although large and powerful, are difficult to use and require a good grounding in spatial theory to exploit their full potential. While computer-aided learning (CAL) techniques offer the ideal means of satisfying the rapidly growing demand for instruction in the use of these systems (Alessi and Trollip 1985), traditional approaches to tutor construction appear to be of little use in meeting the special requirements of an application based on complex graphical manipulation (Rhind 1988).

THE ARCDEMO PROJECT: A PERSPECTIVE

The ARCDEMO project began as a response to the wide interest in the instalment of ARC/INFO at Birkbeck College, University of London, one of the first geographical information systems in Europe. Since the purchase was partly funded by the UK Economic and Social Research Council (ESRC) an early attempt in 1984 was made to make a demonstration and tutor available by interactive logon over JANET. Subsequently, ARCDEMO was installed at a number of sites in the UK and abroad.

Development of ARCDemo

ARCDemo was written in both DEC VAX command language (DCL) and FORTRAN, and runs under the VMS operating system (Green 1987). The system produced graphic output for Tektronix (or compatible) graphics terminals and used plot files to generate the graphics used in the displays. The main subjects covered included map editing, network analysis, projection change, buffering and map overlay. The names and affiliations of those who accessed the system were stored, and mail could be sent by users directly to the system manager.

Extending and improving ARCDemo

ARCDemo suffered from a number of shortcomings, most of them system dependent. Thus, the use of ARCDemo on a heavily used multi-user machine meant that the system suffered from variable response rate. Also since ARCDemo was mostly used over a network with a restricted transfer rates and variable reliability, access to the system could be a problem.

In addition, ARCDemo although a menu driven system suffered from a relatively inflexible structure. For example, although the various sections of the demonstrator could be read in any order, it was not possible to jump into or out of the middle of a section. The sequence of text and graphics could also only be read forwards, and to re-read a single page the whole section had to be read again. Similarly, graphics are stored as single images and cannot be merged or overlaid, and since the screen is cleared between each graphic no visual effects can be used, for example, to dissolve images into each other. Thus, the graphics act simply as colour illustrations.

ARCDemo was also limited by the speed with which graphics could be generated and displayed. This made it necessary to compress a number of different concepts onto just a few images, sometimes relying on fairly crude graphics to convey these ideas. For example, the section in ARCDemo which deals with "cleaning" vector data introduces a large number of subjects and terms including map registration and transformation, generalisation by coordinate thinning, node snapping and overshoot error correction in just 3 images.

THE DESIGN AND CONSTRUCTION OF GIST

Whilst the ARCDemo project met the demand for a system to illustrate the capabilities of GIS in general and ARC/INFO in particular, the growing needs of in-house teaching and the explosion of new GIS software required the design and implementation of a more sophisticated tutor. The project to develop a new tutor was therefore initiated in May 1988.

Design criteria for GIST

The design of tutors for geographical information systems presents some significant challenges to the developers of CAL tools. The key problems identified in this project were the human-computer interface (HCI), the need for interactive graphics and an overall flexibility in the design structure to cater for different learning strategies.

Recent research has demonstrated the value of the use of highly interactive systems in the learning process. Systems using window managers, icons, mice, and pop-up menus (WIMP) interfaces have become popular following their development in the Smalltalk-80 project at XeroxPARC (Goldberg and Robson 1983) and successful implementation on the Apple Macintosh. These techniques now represent a new orthodoxy in the HCI for microcomputers and workstations. Recent research by the Gartner Group (1987) showed how computer users were able to learn more quickly and become fully familiar with more applications when using a consistent WIMP interface, rather than a standard command line system. This can be traced to the use of visual cues in the icons, the ability to select options from menus and the execution of commands using the mouse. The learning overheads on the user can be reduced by making the system easy to use: this leaves more time and attention to be devoted to the subject matter.

A second key problem identified concerned the predominance of graphical subject matter required to teach GIS concepts. To illustrate these concepts requires sophisticated graphical capabilities to display maps and the operations carried out on them. However, to exploit fully the advantages of the graphical display the user should also be able to interact with the maps and images presented. Accordingly, a WIMP interface was seen as necessary to simulate real graphical operations and to allow the illustration of multi-step procedures such as digitising using animation techniques.

Thirdly, the development of CAL tools to fulfil a wider range of learning strategies was seen as desirable (Watson 1987). Many of the available tools for building tutors require the mastering of a complex authoring language such as PILOT and tend to enforce a 'forward working' mode of learning (Walker and Hess 1984). These systems are also designed to provide answers to specific problems in a well-defined (and limited) domains. This question/ answer mode of operation was considered inappropriate for the design of a GIS tutor where the knowledge domain is large, rules are relatively difficult to define and users often have limited knowledge of the subject area (Green 1986).

Hypermedia systems seemed to offer the best opportunity to address these problems since they stress the linkages between subjects and formats of presentation, for example in text, static graphics, animation and video (Ambron and Hooper 1988). The successful implementation of these concepts has however only recently become possible with the development of object oriented languages (OOL). In 1987 Apple Computer released 'Hypercard' for the Macintosh (Goodman 1987) which was amongst the first OOL for a microcomputer (implementing the language 'Hypertalk'), and which went much of the way to providing the functions of a Hypermedia tool.

Hypercard combines the functions of a flexible database management system with a high level OOL into an integrated system. The software uses the metaphor of a card "stack" to form a simple and easily understood analogy, and includes

graphical and text tools for the generation of visual materials. The "cards" in the stack contain text and graphics which may be unique to one card or be shared by all the cards in a stack. Cards are linked using "buttons"-active areas of the screen which when pointed to by a user with the mouse will carry out some action specified by an accompanying program or "script". Text information is stored in "fields" whose font, size and style may be defined; each field can also be made to scroll in order to store large amounts of text on a single card.

In addition each card is made up of a number of layers which contain fields, graphics and buttons. Some layers belong only to a single card whilst others may belong to all or part of a stack. Items stored on the "background" layer can be occluded by foreground items which belong to a single card, and items may be made transparent so for example text or graphics can be viewed through a text field or button. The system is event-driven where the user initiates actions to invoke a script, which may be associated with any of the "objects" discussed. The script is activated upon the receipt of a "message" sent by the event which is passed along a hierarchy of objects by a message "handler" until it reaches the appropriate object. Hypercard can also be extended by adding external commands and functions compiled in the Pascal or C programming languages.

The combination of Hypercard and the Macintosh with its WIMP interface seemed to represent the ideal environment for the development of an interactive GIS tutor. The ability of the system to display and integrate graphics (including animation) was attractive as was the opportunity to establish an "open" tutor with cross-referencing links and powerful search capabilities. Hypercard is also easy to use and understand and offered the possibility to develop a tutor quickly and without the extensive testing required by other software. The main disadvantages of Hypercard are that at present it has limited colour facilities and will only run on the Macintosh, although other less sophisticated Hypertext systems do exist for other makes of computer.

Construction and layout of GIST

Having selected Hypercard as the development tool for the tutor the most significant problem to be overcome was the structuring of the information to be presented. The design of the tutor followed a similar line to ARCDemo in that individual topics were split into separated sections. There was, however, a wider range of topics which could be covered with a more flexible tutor and so many of the sections were split into sub-sections. The range of graphical effects and animation capabilities of HyperCard allowed some new opportunities to give greater visual impact to the tutor. The tutor was designed to run exclusively "mouse-driven" and required no commands to be entered at the keyboard.

One of the main advantages of constructing the tutor within HyperCard was that little formal design was necessary; individual sections of the tutor could be developed independently and then linked together using buttons or

programmed links. The only important design decision, which was taken at an early stage in the tutors development, was to split individual sections into separate HyperCard "stacks" controlled by an introductory stack. This increased the ease with which new sections could be added to, edited and reorganised.

On execution the tutor opens an introductory stack which provides access to the tutor, help facilities and other features. The tutor displays a title page moving to a "welcome" page describing the main options offered by the tutor together with credits. These options include: an "introduction"; a "table of contents"; a "map" of the tutor; a GIS trade directory; a bibliography of GIS publications; and an "encyclopedia". The introductory section is provided for new or inexperienced users and contains several cards or pages describing how to move through the tutor; the structure of the sections included; the actions taken by the various buttons; and how to use the "table of contents" and "map" facilities.

The tutor "contents" facility is equivalent to the table of content in a book. A one line description of each of the main tutor sections is provided together with a button linked directly to it. If a section is selected by 'clicking' on the appropriate button, then a title page for that section is displayed which contains a brief description of its contents. Section title pages also usually contain buttons linked to sub-sections (see, for example, figure 1 below) and also other main sections, the contents and the map.



SEARCH



Data retrieval is the ability of a geographical information system to browse, search and selectively retrieve information from a geographical data base. Map features can be retrieved either on the basis of an attribute, for example, selecting census areas on the basis of population size, or graphically by applying a spatial operator. A graphical operator, for example, might be used to find all population centres within a specified distance of a main road or motorway. The concept of selecting features by applying logical or Boolean operators is fundamental to both these techniques.

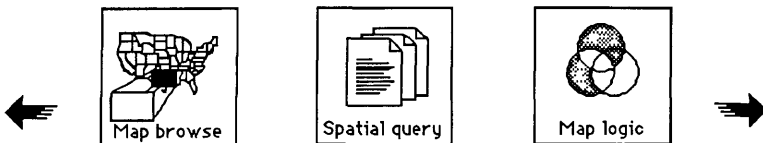


Figure 1 Title page for the "Search" section

The "map" facility (see figure 2 below) provides a visual index to the tutor. The map depicts the structure of the tutor sections and sub-sections and also provides instant access to any of these components. All the sections and sub-sections illustrated in the "map" are also buttons, which provides a browse facility for users with a specific interest. The "map" is accessible from almost any point within the tutor to facilitate this process.

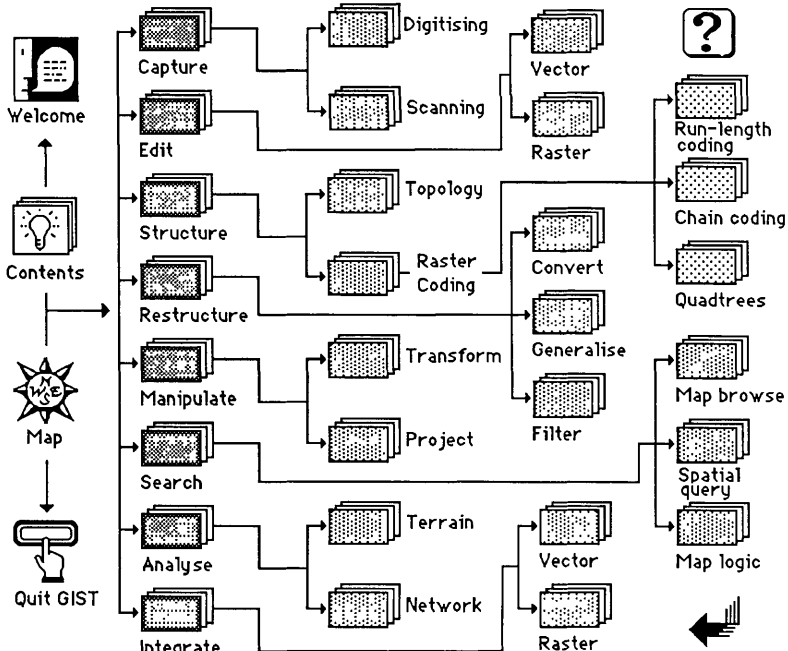


Figure 2 Map of the sections and sub-sections in GIST

The trade directory and bibliography options provided on the "welcome" page are linked to databases which supplement the information given in the tutor. The trade directory provides information on approximately 35 GIS systems currently available in the UK. The address of the company marketing the system is provided together with a brief description of the software including then form of attribute and graphical storage used, and some basic information on hardware requirements. The bibliography lists recent publications which should be both easily available and understandable to those with a limited knowledge of GIS and is arranged by sections covered in GIST. The bibliography also provides details of journals, newsletters, etc. which often carry articles on GIS and related topics (eg. Mapping Awareness, The International Journal of Geographical Information Systems and GIS World) and gives the publishers addresses.

Finally, the encyclopedia is a further feature provided for those users who wish to look up specific terms of interest. Once a particular subject has been accessed the user has the

option to return to the encyclopedia or to continue to remain in that section of the tutor.

In a typical session where the user decides to work through the tutor section by section, GIST maintains a record of the sections and sub-sections completed, flagging them to the user by highlighting the buttons on the section header card. This record is maintained both in this mode or if the user chooses to browse through the sections using the map or the encyclopedia. In order to facilitate browsing or following a train of thought, cross reference buttons are provided to link associated subjects, for example line thinning in the sub-section on 'raster editing', and vectorisation in the conversion sub-section. This implementation of cross referencing and the immediate access to the map from almost any point would be difficult in a non object-oriented system.

GIST AS A GIS TEACHING TOOL

Core activities in GIS

Few curricula exist for GIS topics at present: the range of books available as introductory texts is limited and often existing courses have developed around staff expertise or have become out of date quickly as (often monthly) software developments accumulate. Accordingly the authors decided to design their own approach from first principles, based on an assessment of GIS functionality in spatial operations rather in a software dependent fashion. Thus, the coverage of GIST in version 1.0 is based around the general principles of spatial data handling in a computer system. However, the modular design is easily extendible and further objectives of the project include the addition of application modules, for example illustrations of the handling of remote sensed imagery. The modules defined are described in the sections below.

Data capture is divided in sections dealing with vector and raster methods. In each case, an explanation of the concept is followed by a description of the various techniques. Vector data capture, for example, is illustrated by the processes of manual digitising. The process of encoding a map is made more apparent by animating a digitising session showing the digitising cursor moving over a map sheet along with the generation of line and node features. Raster scanning is used to illustrate the concept of raster data capture, and the technique is shown reducing a map to a pixel image.

Editing is also illustrated using examples applicable to raster and vector data structures. Vector data editing uses the output derived from the vector data capture section to show how simple errors associated with manual digitising can be corrected. In this example, overshoots, undershoots and misplaced points are highlighted and then techniques for solving them described by 'clicking' on the error. An explanation of the problem and how to solve it is then provided, and corrected. Raster editing illustrates gap removal, line thinning and stray pixel removal using a similar technique.

The processing of structuring digital map data (storing the data in a form suitable of analysis and easy retrieval) is also subdivided by vector and raster data types. The concept of polygon topology is illustrated by warping a simple figure to show how topology overall shape are unrelated: the connectivity and adjacency of the figure are shown using simple animation. Raster coding techniques shown include run-length coding, chain coding, and the production of a quadtree from a simple region defined on a grid. In this example, as the figure is recursively sub-divided the quadtree is built alongside and the user can move back and forth to examine the process in detail.

A section entitled restructuring covers the concepts lying behind map generalisation and filtering techniques. The Douglas-Peucker coordinate thinning algorithm and the calculation of a low-pass filter, for example, are illustrated using a stepped animation sequences. This section also deals with techniques for spatial data structure conversion including raster-to-vector and vector-to-raster processing.

Map sheet manipulation is divided into sub-sections dealing with map transformation and projections. Transformations-rotation, translation, scale change and warping are illustrated by animating these operations when applied to a simple grid. The fundamentals of map projections are also explained and examples generated by the ARC/INFO system are used to show the effect of changing the projection of a map of the USA.

The concepts of specifying a search and the retrieval of information using a GIS are covered in 3 sub-sections dealing with map windowing operations; Boolean logic for map operations; and spatial query. These sections rely on a large amount of user interaction including question/answer sessions, the construction and application of selection criteria to a database containing selected population statistics for railway stations in SE England.

Network and terrain analysis are chosen as being typical of the types of spatial analysis performed by a GIS. A step-by-step description of the fundamental algorithms (such as finding the shortest path through a network) is provided along with several animated sequences illustrating the building of terrain models (including a fly-by of Mount St. Helens!) which is becoming an increasingly important area of GIS research (Raper 1989).

The final section, which deals with map integration, helps to illustrate some of the important differences between the raster and vector data models. Some of the topics covered include: the concept of integrating attribute information; the problems of "sliver" errors generated by vector-based overlay; and the concept of applying "masks" to assist raster integration. Finally, the the section culminates with an animated illustration of vector-based map overlay to select a region fulfilling multiple selection criteria.

Using GIST in a short course programme

The topics above when taken together define a set of core activities within GIS which can be used as a short curriculum. The tutor is already in use in the bimonthly Short Course programme at Birkbeck College (funded by the UK Dept. of Trade & Industry PICKUP initiative), where the objective is to provide an introduction to GIS and spatial data handling.

In order to be useful in teaching GIST contains a report generation facility which allows the instructor to check on the progress of the student through the system. This report is also used to track the interests and aptitudes of the students in order to indicate where to improve and develop the coverage. In teaching the short courses the authors have found that use of GIST individually by a small class creates a productive learning environment since the instructor can respond to individual queries as they arise. The students can also customise their own notes by printing out the cards using Hypercard's own report printing facility, and then annotating them as appropriate.

CONCLUSIONS

A new GIS interface?

A strategic objective lying behind the development of GIST is a wider design approach to the improvement of the interface to GIS. In general GIS systems are difficult to use and a significant learning overhead to most systems lies in training associated with the operating system and the command structure. It is considered that the techniques used in GIST may provide the basis for the design of a generic spatial language interface. Since Hypercard can expand commands to remote computers using communications software such as Sequelink, it is anticipated that some of the tutors' demonstrations could be implemented by handshaking with a real GIS and returning the data to GIST. This has provided the blueprint for the development of a spatial language named UGIX and is discussed more fully in Rhind, Raper and Green (1989) (this volume).

Future developments

A number of UK Higher Education Institutions are currently using GIST in their teaching programmes as an introduction to spatial theory and GIS. GIST version 1 represents the core of a GIS tutor which it is hoped will provide a basis for a wider range of Hypermedia materials currently under development, some of which are to be funded under the UK Economic and Social Research Council's Regional Research Laboratory Initiative.

The authors consider that both the underlying principles and implementation of GIST represent a considerable advance on existing learning materials in GIS. However, in this fast moving industry the requirements for training materials are constantly expanding, and a considerable challenge remains to meet these further needs.

ACKNOWLEDGEMENTS

The authors wish to thank Apple Computer (UK) for the loan of a Macintosh II computer with which to undertake this work.

REFERENCES

- Alessi, S.M. and Trollip, S.R. 1985, Computer based instruction methods and development, Prentice Hall.
- Ambron, S and Hooper, C 1988, Interactive multimedia, Microsoft Press.
- Gartner Group 1987, Apple Computer and its Macintosh: product report, available from Apple Computer.
- Goldberg, A and Robson, D 1983, Smalltalk-80: the language and its implementation, Addison-Wesley.
- Goodman, D 1987, The complete Hypercard Handbook, Bantam Books.
- Green, N.P.A. 1986, User/ system interfaces for geographical information systems, Report 4 NERC Remote Sensing Special Topic: The conceptual design of a GIS for the Council. Birkbeck College, University of London.
- Green, N.P.A. 1987, Teach yourself geographical information systems. The design, creation and use of demonstrators and tutors. International Journal of Geographical Information Systems, Vol.1, pp. 279-290.
- Raper, J.F (ed.) 1989, Three dimensional applications in GIS, Taylor and Francis.
- Rhind, D.W. 1988, A GIS research agenda. International Journal of Geographical Information Systems, Vol. 2, pp. 23-28
- Rhind, Raper and Green 1989, First UNIX then UGIX, (this volume).
- Walker, D. and Hess, R. 1984, Instructional software: principles and perspectives for design and end use, Wadsworth Publishing.
- Watson, D. 1987, Developing CAL: Computers in the Curriculum, Harper and Row.

DEMOGIS MARK 1: AN ERDAS-BASED GIS TUTOR

David J. Maguire
Department of Geography
and
Midlands Regional Research Laboratory
University of Leicester
University Road
Leicester
LE1 7RH
England

E-Mail JANET MAG@UK.AC.LE

ABSTRACT

There has recently been a number of calls by influential bodies for the development of computer-based GIS tutors. This paper reports on the progress in the continuing development of such a tutor called DEMOGIS. The tutor is based on the ERDAS Image Processing and GIS software package. It comprises a linear sequence of frames that deal with the principles of GIS and the application of GIS for locating land potentially suitable for mining and quarrying in the Charwood Forest region, England. The limitations of DEMOGIS are discussed along with possible future developments.

INTRODUCTION

The relatively new field of Geographical Information Systems (GIS) is developing extremely rapidly at the present time and there is great interest in all aspects of it in governmental, commercial and academic spheres. The recent origin and rapid rate of development of the field have caused a number of problems, not least of which are the worldwide shortage of skilled staff and the lack of information outlining the fundamental principles and applications of GIS (Green 1987, Rhind 1988, Maguire 1989). This has led a number of influential bodies to call for initiatives in GIS education and training (DoE 1987, NERC 1988). The Committee of Enquiry into the 'Handling of Geographic Information', established by the British Government and chaired by Lord Chorley, for example, made nine recommendations concerning education and training (DoE 1987). Recommendation 50 stated "As part of the measures aimed at increasing awareness of Information, familiarization with geographic information technology should be encouraged throughout the education system" and 57 stated "The Department of Education and Science and the MSC [Manpower Services Commission] should encourage the development of computer-based interactive packages for teaching and home based learning in the handling of geographic information".

In addition to answering these calls for general initiatives in education and training, there were a number of significant practical reasons why we should develop a GIS tutor. The Department of Geography and the closely associated Midlands Regional Research Laboratory (MRRL), funded by the Economic and Social Research Council, are heavily involved in GIS education and training. In the Department's undergraduate programme there is a final year option in GIS taken by up to 20 students. The Department also runs a taught MSc in GIS which attracted 9 students in 1988/89. As well as this, the MRRL frequently arranges short courses in GIS, targeted at people in commercial and governmental organizations, and is frequently asked to arrange demonstrations of its activities at conferences and exhibitions and for potential clients. These activities all place a great burden on University and MRRL staff. The Department also has a long tradition of work in Computer Assisted Learning (CAL) and, given current interest in GIS, attention was naturally focused on this area.

This paper describes progress in the continuing development of a computer-based GIS tutor, called DEMOGIS, which was designed as a response to the calls for initiatives in GIS education and training, as a contribution to education and training programmes and as a test bed for applying CAL ideas to the GIS field. The remaining part of this paper consists of four main sections. The first section examines the design considerations for a GIS tutor, the second discusses why ERDAS was used for the development of DEMOGIS, the third presents an overview of the tutor and the fourth contains discussion and conclusions.

DESIGN CONSIDERATIONS

When designing the GIS tutor, DEMOGIS, a number of important requirements were identified. The tutor had to: be robust because it would be shown to potential clients and students; be capable of use by novices with little or no computer or GIS experience; require minimum user input because it would be used at conferences and exhibitions as a publicity device; run quickly and be entertaining in order to capture users' attention; illustrate something of both the principles and applications of GIS; incorporate both raster (e.g. satellite) and vector (e.g. thematic map) data, in contrast to the best known tutors, ARCDemo and ECdemo, developed by Green *et al.* (Green 1987), which only incorporate vector data; use examples relevant to the local (Midlands) area because of the regional dimension to the MRRL and the availability of relevant data in machine readable format; be produced quickly within a timescale of a few months because of impending commitments; be portable so that it could be transported to conferences and exhibitions.

THE ERDAS SOFTWARE

ERDAS (Earth Resources Data Analysis System) is a computer-based system for generating, managing, processing and displaying geographical data (ERDAS 1987) which is available for a wide range of micro- and mini- computers. The work described here was undertaken on a WYSE PC (an IBM PC/AT compatible) fitted with a No.9 framestore board. The ERDAS software is modular and has been under development since 1979. Briefly, there are modules for Image Processing (IP), Geographical Information Systems (GIS), Polygon Digitizing and Topographic Modelling. An interface to the dBASE data base management system is also available.

ERDAS was selected as the development tool for a number of reasons. The short time scale dictated that it was not possible to write the required software from scratch. In any case, ERDAS has good graphics and a number of useful facilities for creating demonstrations (see next section). ERDAS runs on a microcomputer and, therefore, any tutor developed using it would be easily portable. Although ERDAS is a raster-based system, with very good facilities for handling remotely-sensed data, it has the capability to input and integrate vector data using the Polygon Digitizing module. Far more important than any of these reasons, however, was the fact that the Department of Geography owned an ERDAS system and had some experience of its use.

OVERVIEW OF DEMOGIS MARK 1

The DEMOGIS (DEMonstartion Of a GIS) tutor was created by the author and Mr Trajco Mesev in 1988. It aims to introduce newcomers to some of the basic principles and applications of computing and GIS. Specific objectives include: an explanation of the term GIS; an outline of the basic elements of GIS; an overview of the fundamental principles of GIS, including a discussion of their functionality; presentation of an example using real data to illustrate some of the potential applications of GIS; and an introduction to the ERDAS GIS software.

DEMOGIS actually comprises a collection of 38 frames (Figure 1) which are automatically replayed in a linear sequence. It begins with a title frame (Figure 2), a short introduction to its scope and content and a brief explanation of what is meant by the term "Geographical Information Systems" (Frames 1-2). The main body of DEMOGIS is made up of two parts. The first is concerned with the principles of GIS (Frames 3-14) and the second (Frames 5-38) deals with an application.

1	Opening title
2	Contents, definition of a GIS
3-14	PRINCIPLES OF GIS
4	Components of GIS
5	Data collection
6-8	Data storage
9	Data manipulation
10-14	Data display
15-37	AN APPLICATION OF GIS
16-18	Location of study area
19	Summary of data layers
20-24	Maps of data layers
25	Search criteria
26-36	Data extraction and overlay
37	Suitable land
38	Closing title, names and address of authors

Figure 1. Contents of DEMOGIS Mark 1

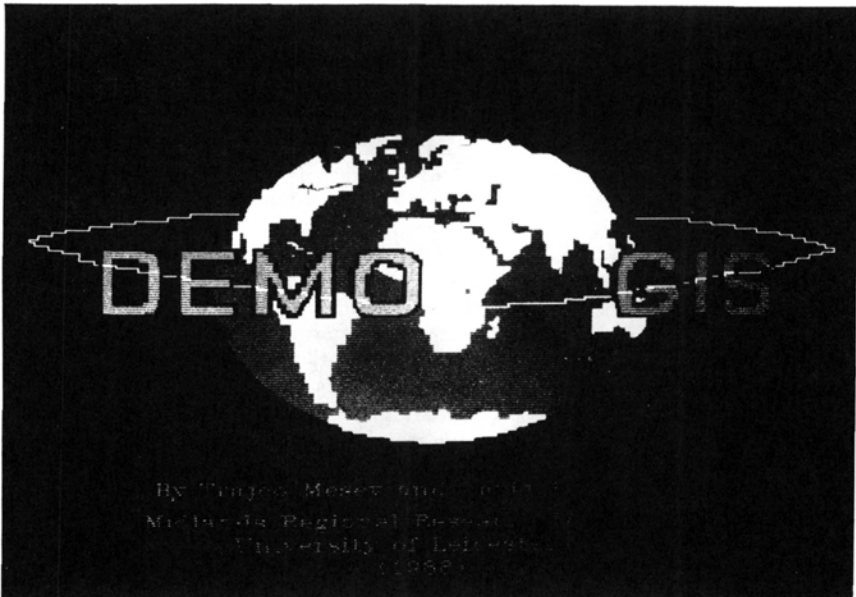


Figure 2. Opening title page.

Part 1: Principles

GIS deal with both geographical locational and statistical data. The locational data describe geographical features such as hospitals and soil pits (points), roads and rivers (lines) and postcode sectors and geological units (areas). These provide a spatial reference for statistical data such as hospital sizes, river flows and rock types.

Geographical locational data may be collected and stored in a number of different ways of which the most popular are the vector and raster systems. Each of these has its merits depending on the specific application.

The most commonly used data collection devices are, for vector data, the digitizer and, for raster data, the line scanner.

5

014 26

Figure 3 Data collection frame.

Part 2: Application

- A: Solid geology - extract land suitable for mining and quarrying (coalfield, igneous, sandstone, limestone, syenite and dolerite)
- B: Agricultural land classification - remove land unsuitable for mining and quarrying (grade 1 and 2 land)
- C: General land use - remove land unsuitable for mining and quarrying (water bodies, urban areas and conservation woodland)
- D: Transport network - only consider land which is easily accessible to mining vehicles (within 150m of an existing road or railway)
- E: Past and present mining and quarrying - consider only land not previously mined or quarried

25

Figure 4. Data layers and search criteria used in the example application.

Part 1 begins with a description of the key components of GIS and then considers data collection, storage, manipulation and display. The frame on data collection (Frame 5, Figure 3) identifies the fact that GIS deal with both geographical locational and statistical data. Some point, line and area examples of both of these are presented. The terms digitizer and scanner are introduced. The three frames which deal with data storage (6-8) outline the differences between the vector and raster structures, using two diagrams, and give some examples of GIS software packages which utilize these structures. Frame 9 lists the main types of data manipulation operations which GIS can perform, namely, cartographic operations, data integration, feature measurement, spatial searching and statistical analysis. The final five frames in Part 1 (10-14) show some examples of how ERDAS can display data in map form. The examples show data on population density (taken from the 1981 Population Census of England and Wales small area statistics), at local authority ward level, for the Charnwood Forest region, Leicestershire (the study area chosen for Part 2 of DEMOGIS).

In Part 2 an application is presented which shows how a GIS can be used to locate land potentially suitable for mining and quarrying in the Charnwood Forest region. The study area was chosen because it falls within the geographical area of responsibility of the MRRL, because of the varied nature of its human and physical geography, because of the problem of the local impact of mining and quarrying, and because of the availability of suitable data. The Charnwood Forest region is situated in North West Leicestershire in the Midlands of England and includes the towns of Loughborough, Coalville and Shepshed. It covers an area of approximately 180 square kilometres, the majority of which is rough grassland and heath, with a little woodland. The highest point is Beacon Hill at an altitude of 248 m. The Region has extremely diverse geology and there is a long history of coal mining and quarrying of igneous, sedimentary and metamorphic rocks. The Region is widely used for recreation, but there is considerable pressure to extend current mining and quarrying activities. The opening frames in Part 2 (15-24) introduce the study area and the data layers in the data base (these are listed in Figure 4). The data layers include both vector (e.g. Geology, Figure 5) and raster (e.g. General land use (derived from classification of a Landsat Thematic Mapper satellite image); Figure 6) data. Examples of point (e.g. Mining activity), line (e.g. Transport network, Figure 7) and area (e.g. Agricultural land) data are also included. The user is shown how each of these data layers can be searched using selection criteria (Figure 4) to extract the relevant categories of land use. The relevant categories of land use are then combined by overlay to give the land potentially suitable for mining and quarrying (Frame 37, Figure 8).

Each of the thirty-eight frames in DEMOGIS had to be compiled separately using the ERDAS software and then stored on the hard disk in ERDAS dump screen format. When the tutor is run, the frames



Figure 5 Geology data layer used in the application.



Figure 6. General land use data used in the application.

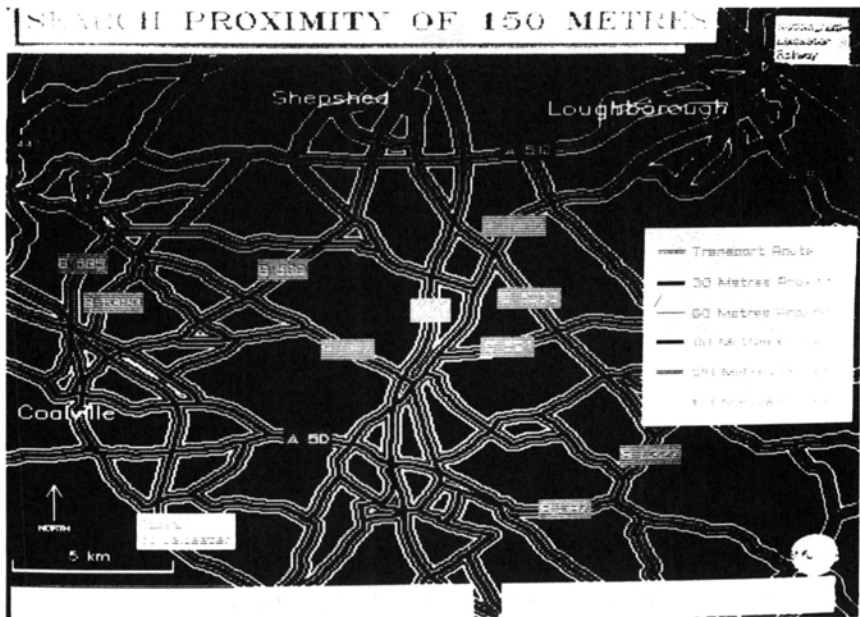


Figure 7. Transport data layer used in the example after creating a 150 m buffer around routeways.



Figure 8. The resultant land identified by extraction and overlay of the data layers.

are displayed in a linear sequence, with the capability to cycle round to the beginning after the last frame. The creation of these frames required completion of a number of separate tasks. The mining and quarrying example application uses real world data and attempts to portray a realistic application of a GIS. Each of the thematic data layers (A,B,D and E - see Figure 4) had to be digitized from appropriate paper maps. Digitizing was undertaken using a Calcomp 9100 semiautomatic digitizer. Because of the limitations of the ERDAS digitizing software (only entity by entity digitizing is available), special software was used for topological digitizing and the digital data were converted into ERDAS format using a FORTRAN program. The maps used for digitizing varied in scale from the 1:25,000 Ordnance Survey topographic maps, from which the transport and mining and quarrying data were collected, to the 1:63,360 British Geological Survey map, from which the solid geology data were collected. The satellite data pertaining to the study area had to be extracted from an appropriate image of the area (collected on 8 July 1984) and then geometrically corrected to the National Grid and classified to give a representation of the general land use of the area. Programs within the very good ERDAS Image Processing software module were used for these operations. Once the basic data had been prepared, spatial processing was undertaken using the programs within the ERDAS GIS software module. This involved extraction of the appropriate land use types from each data layer and then overlay, as described above. In the tutor itself, each of the frames showing the data layers and spatial processing operations were embellished with titles, explanatory text, legends etc., using the good annotation facilities in ERDAS.

The annotation programs were also used to create all the text and graphic frames which make up the remainder of DEMOGIS. In creating these, extensive use was made of the ERDAS symbol library facility. This allows symbols, such as the DEMOGIS logo (Figure 2) to be placed anywhere. at any size, on any frame.

DISCUSSION AND CONCLUSIONS

Throughout the development of DEMOGIS, educational issues have always been given priority. In order to maintain the users interest, DEMOGIS uses a mixture of 15 frames of text, 6 frames with graphics and 17 with maps. The text only frames were kept to a minimum and were limited to a maximum of 150 words and used bright colours. To emphasize the integrated layout and organization of the tutor, where possible, a standard screen layout was adopted (compare, for example, Figures 3 and 4) and each part has been colour coded with a separate background colour: the Introduction is mid-blue, Part 1 is light-blue and Part 2 is green. The spatial processing operations of extraction and overlay are shown step by step with intervening text to explain each step.

DEMOGIS is still under development and so far we have only progressed to a Mark 1 version which has a number of limitations. DEMOGIS relies heavily on the ERDAS software for the displaying the tutor. The frames are stored in ERDAS dump screen format which, although it allows rapid display of files, utilizes a relatively inefficient storage structure. Most of the frames in DEMOGIS are about 0.7 mb in size and this leads to problems of data storage and transfer. This problem has been partially solved by using a shareware data archiving routine called PKARC (PKWARE 1987) which compresses the whole of DEMOGIS into an archive of only 0.8 mb. Probably the biggest limitation of DEMOGIS Mark 1 is the lack of any animation or user interaction. As it stands, at present DEMOGIS Mark 1 is basically a computer book. It does not make use of the full potential of CAL for user learning by experiment and feedback (Shepherd 1985). It is the intention that future versions should incorporate a higher degree of user interaction. This might include experiments into, say, the effects of different size buffers around the transport network, question and answer sections, optional sections with varying levels of explanation and animated sequences. All these require considerable development and whilst there are some facilities available in ERDAS it may well be that some are beyond its capabilities. It is intended that future versions of DEMOGIS will incorporate other aspects of GIS not covered in Mark 1, for example, the use of data base management systems for handling geographical statistical data, map projections, scale change in spatial data bases and digitizing.

In spite of its obvious limitations DEMOGIS Mark 1 has proved to be very useful and it does meet some of the demands for initiatives in GIS education and training. Already it has been used by undergraduates and postgraduates at the Universities of Leicester and Aberdeen and Chester College, and has been shown to a number of potential clients of the MRRL, both in Leicester and elsewhere at conferences and exhibitions. It satisfies the modest design requirements outlined earlier in this paper. In developing DEMOGIS we have learned a great deal about the

educational issues involved in teaching GIS, about ERDAS and about mining and quarrying in Charnwood Forest region. Hopefully, there will be future editions of DEMOGIS which will alleviate some of its current limitations.

ACKNOWLEDGEMENTS

The author would like to thank Trajco Mesev, for contributing the greater part of the spade work in developing DEMOGIS Mark 1, and Bill Hickin and Mitch Langford.

REFERENCES

DoE 1987 Handling geographic information HMSO, London

ERDAS 1987 User's guide ERDAS, Atlanta

Green NPA 1987 Teach yourself geographical information systems
International Journal of Geographical Information Systems 1 279-90

Maguire DJ 1989 Computers in geography Longman, London 245pp

NERC 1988 Geographic information in the environmental sciences. Report of the working group on geographic information NERC, Swindon

PKWARE 1987 PKARC Version 3.5 PKWARE, Glendale, WI

Rhind DW 1988 A GIS research agenda International Journal of Geographical Information Systems 2 23-28

Shepherd IDH 1985 Teaching geography with the computer: possibilities and problems Journal of Geography in Higher Education 9 3-23

CARTOGRAPHIC ANALYSIS OF U.S. TOPOGRAPHY FROM DIGITAL DATA

Richard J. Pike¹ and Gail P. Thelin²
U.S. Geological Survey

¹Menlo Park, CA 94025; ²Ames Research Center, Moffett Field, CA 94040

ABSTRACT

We abstracted the surface-geometric character of U.S. physiographic subdivisions by image-processing 12 million digital elevations (grid spacing 0.8 km). Topographic homogeneity of the 82 Fenneman sections varies widely, elevation (e) having less dispersion than slope (s): estimates of this spread by the coefficient of variation range from a low 0.10 (e) and 0.62 (s) to a high 1.3 (e) and 3.5 (s). A large shaded-relief image of the conterminous 48 states was created by processing the log-transformed elevations through a photometric function. The 1:2,500,000-scale digital map has far more fidelity and detail than other synoptic portrayals of the nation's topography.

INTRODUCTION AND PHYSIOGRAPHIC TAXONOMY

Much of the geologic and tectonic information that lies encoded within topographic form can be deciphered by image-processing large files of digital elevations through fast computers. The three latter elements together have created unprecedented opportunities for automating the numerical and cartographic analysis of topography. Here we address two current needs. One, a regionally based characterization of U.S. terrain that can stand alone or combine with other data, is approached initially by comparing the existing physiographic subdivisions. The second need is for a large single-sheet portrayal of U.S. topography

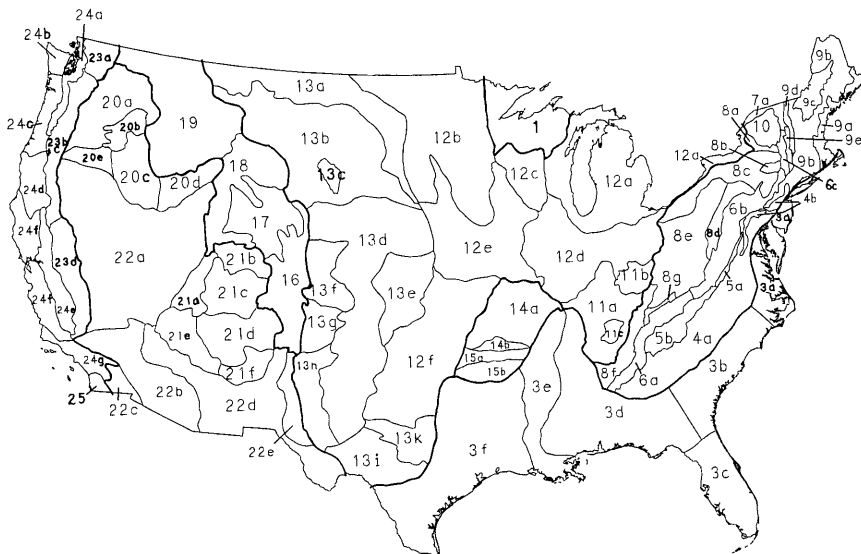


Figure 1. U.S. physiographic divisions (heavy lines), provinces (numbers), and sections (letters) on 1946 map in Fenneman (1931, reprinted 1946). Province 2, Continental Shelf, is not included. The names, omitted here for lack of space, are on the 1:7,000,000-scale map Physical Divisions of the U.S., available from USGS.

that can be used to critique these units and also marks a significant increase in detail and accuracy over past, non-digital, graphics.

A physiographic unit is a distinctive area having common topography, rock types and structure, and geologic and geomorphic history. Such regions of contrasting surface character greatly assist the analysis of continents and large sea-floor tracts. The most widely accepted physiographic units of the U.S. are the eight division, 25 provinces, and 86 sections of Fenneman (1931, 1938). His hierarchical classification (Figure 1), first outlined in 1914, still is useful but both the taxonomy and its implications remain qualitative. The topographic uniqueness evident in most physiographic units has not been expressed numerically, and the presumed homogeneity of land form within them has never been measured or tested. The sole exception is a sampling of slope angle, length, and height within 24 provinces for a study of surface roughness (Wood et al., 1962).

METHOD AND DATA

Raster image-processing, a digital technique developed primarily for handling images reassembled from spacecraft telemetry, has been successfully adapted for manipulating elevation matrices and their derivatives over large areas (Batson et al., 1975; Moore and Mark, 1986; Pike and Acevedo, 1988). Because one pixel = one elevation, the technique is an efficient spatial tool for landform study. The square-grid data structure enables quantitative measures that describe topography to be rapidly calculated, compared, and combined for display as shaded-relief (Figure 2) and other map images or as digital output for further analysis. Our results were generated by subroutines within the Interactive Digital Image Manipulation System* (IDIMS; ESL, 1983) installed on a DEC Inc. VAX 11/780* computer.

The data most suited to our objectives were gathered by the Defense Mapping Agency Topographic Center (DMATC). Contour lines, and later spot heights and stream and ridge lines, on the U.S. 1:250,000-scale topographic sheets were digitized and gridded semi-automatically at 0.01 inch map resolution (200 feet or three arc-seconds on the ground; Mays, 1966). Interpolation between digitized contours accounted for about 5/6 of the resulting data (Noma, 1966). The original DMATC set of over 2 billion elevations is available in $1^{\circ} \times 1^{\circ}$ blocks from the USGS. It has been sampled, regridded, thinned, and averaged (Godson, 1981) down to the more manageable file used here. The 12 million elevations (nearly twice that counting null-value background pixels in the 3750 X 6046 array) are spaced 0.805 km apart (30 arc-seconds on the ground, XY) within the conterminous 48 states.

Both data sets contain mistakes besides those inherent in the original maps. Most errors are systematic. They include flattened hills and inaccurate interpolation between widely spaced contours, the result of a fast but suboptimal algorithm dictated by the small computers of the time (Noma, 1966), and defective splices between data blocks. Less systematic errors include other flattened hilltops and random zero or unduly high elevations arising from unknown causes. Although first read or interpolated to the nearest foot, DMATC data were rounded to 10 m (map contour intervals were 100 feet or more) and may be accurate to no more than 30 m in smooth areas and 50 m in rough terrain.

*Trade names and trademarks in this paper are for descriptive purposes only, and imply no endorsement by the U.S. Geological Survey.

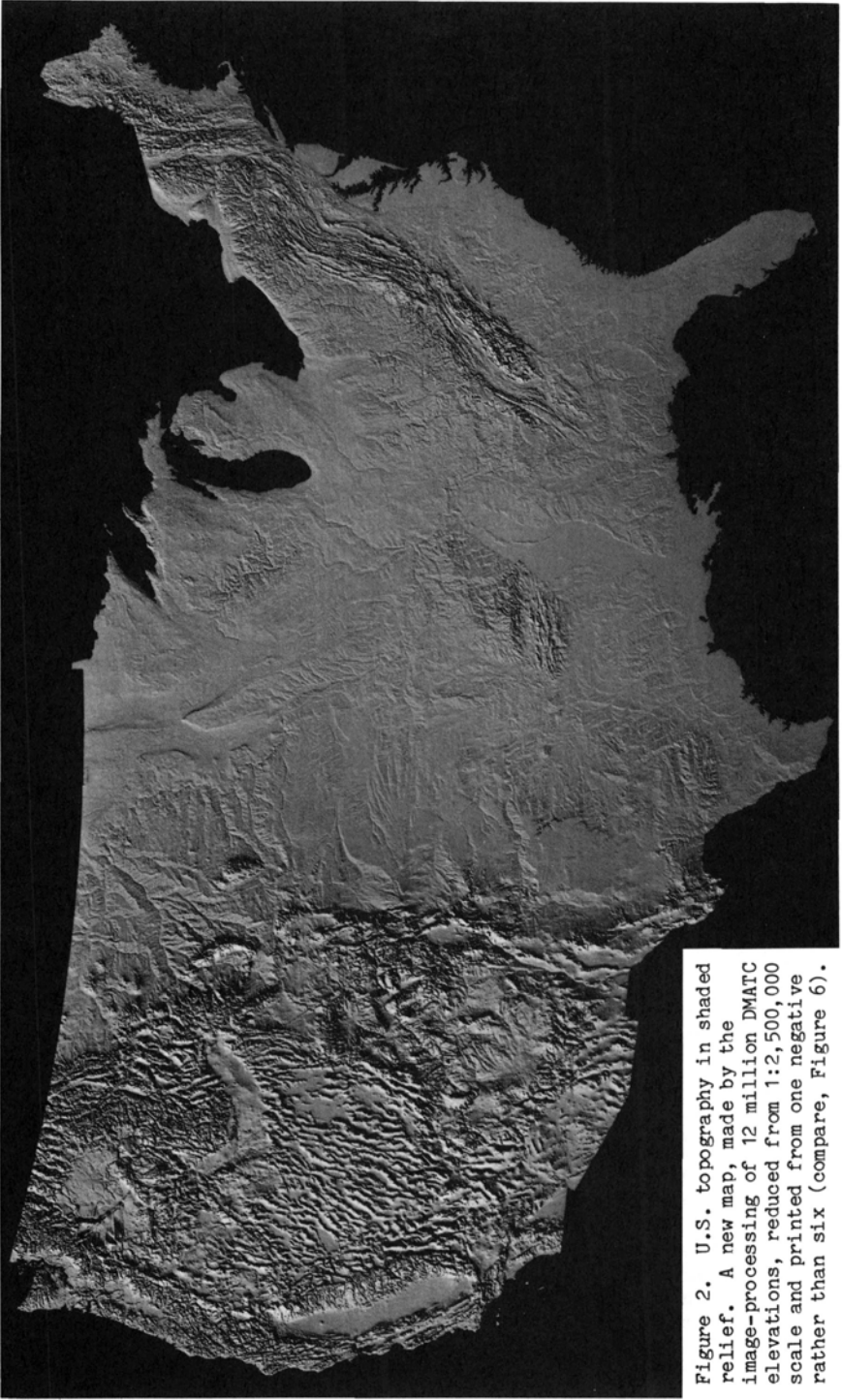


Figure 2. U.S. topography in shaded relief. A new map, made by the image-processing of 12 million DMATC elevations, reduced from 1:2,500,000 scale and printed from one negative rather than six (compare, Figure 6).

THE NEW WORK

We have begun a long-term study of the quantitative properties of regional topography within the U.S. The objective is to translate such verbal descriptions of terrain as gentle, rolling, hilly, steep, subparallel, blocky, and fine-textured into numerical terms that can not be interpreted in more than one way and are easily communicated. The first results are statistical generalizations on elevation and slope angle for 82 of the 86 physiographic sections (no data for sections 2, 7b, 11d; 23b and c were inadvertently combined). These are early steps toward a multivariate parametric characterization, or geometric signature (Pike, 1988), that uniquely expresses topographic form in each unit. We have already begun to synthesize our own units from such signatures (Pike and Acevedo, 1988; cf., Hammond, 1964).

Of the many elements of landform geometry (Hammond, 1964; Mark, 1975; Pike, 1988) elevation and slope are perhaps best suited for the first experiments. Estimates of these, the most straightforward properties, were the very criteria by which Fenneman and other field observers first recognized and summarized the topographic character of the physiographic units. Indeed, many of the Fenneman map boundaries can be reproduced on the basis of whether juxtaposed topography is high or low, rough or smooth (compare Figures 1 and 2).

We generated histograms of elevation (50-foot and 250-foot bins) and slope angle (1° bins) for each physiographic section and computed (to 0.01 units) four measures of central-tendency and dispersion: mean, minimum, maximum, and variance. The calculations were made for the 48-state area by processing all 12 million terrain elevations through IDIMS. First we digitized physiographic section boundaries from the Fenneman map, using Arc/Info* geographic information system software installed on a Prime 9955-II* computer. Upon transferring this vector file to a VAX 11/780 computer, we generated a digital raster image of the boundaries within IDIMS. From the elevation file we created a digital image of topographic slope by means of the TOPOG* algorithm, a polynomial fit to 3x3 neighborhoods of elevations, moved through the data one pixel at a time. Lastly, we calculated statistics for the 82 physiographic sections from the slope and elevation images, using the POLYSTAT* algorithm and the digital image of Fenneman's map lines.

RESULTS

The 82 elevation histograms computed by IDIMS in a single pass (e.g., Figure 3) provide the simplest numerical comparison of the terrain units in Figure 1. Many of the distributions are skewed. Asymmetry of elevation is a basic descriptive property of topography; it should not be removed or adjusted (by transforming the raw data), save for purposes of graphic presentation. Distributions often are bi- or multi-modal, and accordant summits and dominant elevations can be identified readily on many of them. Mean heights of the 82 sections range from 40 feet (Embayed Atlantic Plain, 3a) to 8973 feet (Southern Rocky Mountains, 16); the median for all section means is 1550 feet.

The histograms and statistics of elevation have proved essential for detecting errors and evaluating the accuracy of boundaries of many physiographic sections. Nonsystematic errors are easily recognized as unexpected maxima and minima and as isolated values on histogram tails for regions where such values are clearly impossible. Most of these errors correspond to equally aberrant values in histograms of slope angle, each bad elevation resulting in several bad slopes.

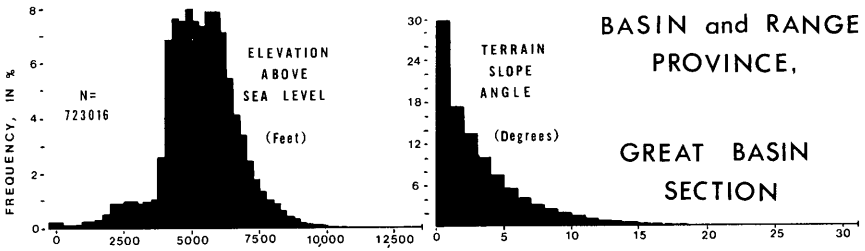


Figure 3. Histograms of height (L) and slope (R) for one of 82 Fenneman sections (22a). Elevations at 1% frequency are lower (southern) basin floors at 2000-3500 feet; those at 7-8% are higher (northern) basins at 4000-6000 feet. Slopes on some mountain ranges exceed 30° , but low values prevail overall.

Many other high elevations, evident in the histograms as unusually long tails, are not incorrect but rather belong to neighboring physiographic units that differ strongly in height. We checked several of these suspect sections (California Trough 24e, Salton Trough 22c, St. Lawrence Valley 7a, and Puget Trough 24a), by overlaying the Fenneman map on a topographic map of the U.S. We observed in every case that boundaries as currently drawn occasionally intersect mountain ridges that protrude into the section from a neighboring unit, rather than skirting them as they should.

Slope-frequency distributions, which are smoother than those of elevation for the Fenneman sections, are never multimodal, even in the dichotomous terrain of the Basin and Range province (Figure 3). All histograms of slope are highly and positively skewed. Mean slope angle as calculated currently (we have not yet transformed the data to improve statistical validity) on the 1.6-km slope samples ranges from 0.004° (Florida 3c) to 10.6° (northern Cascades 23a). This 1000X difference nicely summarizes, quantitatively, results from the most disparate slope-forming processes within the conterminous 48 states. Median slope for the 82 regions is about 1.35° .

Groupings of map units in slope/elevation space (Figure 4) affirm most of the Fenneman hierarchy. Sections (dots in Figure 4) within the same province link to form polygons. Not only are most polygons quite compact, but the provinces overlap remarkably little. Four sections lie abnormally far from other province constituents: New England Seaboard Lowland (9a), California Trough (24e), Wisconsin Driftless area (12c), and Salton Trough (22c). The two Ozarks sections (14a,b) differ markedly. Fenneman's eight divisions (Figure 1) also are evident in Figure 4 as fairly distinct clusters of slope/elevation polygons. Only the Laurentian Upland (Superior Upland, province 1) resembles another division, the Interior Plains (prov. 11-13).

Mean slope in the U.S. is a weak log-log function of elevation (Figure 4). Although such covariance has long been suspected, the nature of the relation and its anomalies now can be determined exactly. For example, the Western Lakes (12b) and three Great Plains sections (13d-f) lie at a much higher elevation than is normal for such smooth terrains. Three Pacific Border sections (24a-c) are much rougher than would be expected for such low-lying areas. These atypical values contribute to unique characterizations, or signatures, for the sections and pose intriguing problems for geologic interpretation.

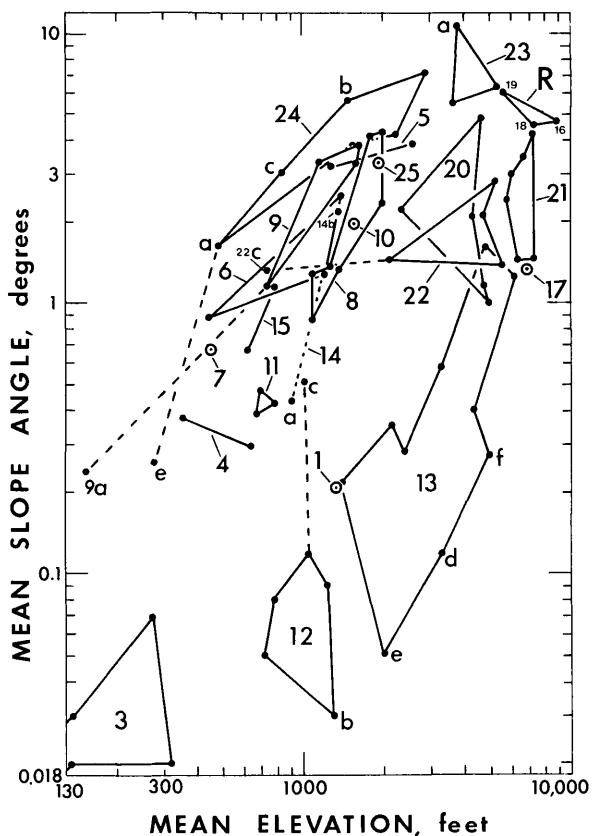


Figure 4. Groupings of physiographic divisions in the conterminous U.S., according to average steepness and height above sea level, computed from 12,000,000 digitized DMATC elevations. The 82 Fenneman sections (dots, small letters) form compact polygons (numbered provinces). Dashes join atypical sections to provinces. R is Rocky Mountain system. Province 3 extends to lower values (2 sections: Florida, Mississippi Alluvial Plain).

The topographic homogeneity of U.S. physiographic subdivisions varies widely (Figure 5). Our estimates, from a statistic of relative dispersion -- the coefficient of variation (C_v , standard deviation/mean), are provisional. Software limits excluded the 13 smoothest sections and the skewed slope distributions have not been transformed. Values of C_v available for slope range from 0.6 (Pacific Border, 24, and Cascades, 23) to over 3.0 (California Trough, 24e); those for elevation vary from 0.10 (Harney section, Columbia Plateau, 20e) to 1.3 (Salton Trough, 22c).

Slope dispersion may be a very weak, inverse, log-log function of elevation dispersion (Figure 5). The C_v relation only vaguely resembles that for slope/elevation. Although sections cluster by province and provinces by division, as in Figure 4, the resulting polygons overlap more. Median C_v s are about 0.33 (elevation) and 1.26 (slope), values of relative dispersion that best typify the

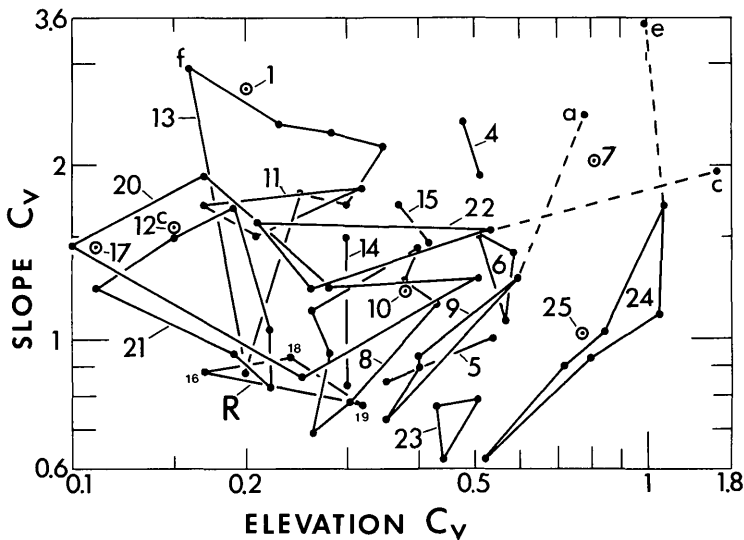


Figure 5. Varying topographic homogeneity of physiographic units in the conterminous U.S. Relative dispersion of slope, C_v , as a function of elevation dispersion, C_v , for 69 Fenneman sections (same label conventions as in Figure 4). C_v , the coefficient of variation -- standard deviation / mean, is dimensionless.

Adirondacks (10), the Appalachian Plateaus (8), and some sections of the Columbia Plateau (20). Most Pacific Border (24) sections and the Salton Trough (22c) are far above average in heterogeneity of elevation. Slopes in the Superior Upland (1) and St. Lawrence Valley (7) provinces, plus Colorado Piedmont (13f), New England Seaboard Lowland (9a), and California Trough (24e) sections, are more heterogeneous than average. Some of these anomalies may diminish upon revision of the Fenneman taxonomy and refinement of section borders.

A NEW MAP OF SHADED RELIEF

Analytical hill-shading, the portrayal of topographic form by mechanical techniques (Brassel, 1974; Horn, 1981), was adapted for computer automation and elevation data in raster format by Yoeli (1967). Batson et al. (1975) made the first shaded-relief images of regional extent, from the DMATC terrain data tapes, and Arvidson et al. (1982) published the first image of the continental U.S., albeit at 1:30,000,000 scale, from the 30-arc-second data. Broad-scale shaded-relief maps of South Africa and Australia followed (Moore and Simpson, 1982; Lamb et al., 1987), while in the U.S. experiments continued to improve the technique (USGS, 1986; Scholz et al., 1987).

We made a wall-sized shaded relief image of the conterminous 48 states by processing the entire file of 30-arc-second elevations through IDIMS. After creating two files on the VAX 11/780 (data were supplied in eastern and western U.S. halves), we registered each on Albers equal-area projection, using bilinear interpolation, to yield 0.805-km pixels. Once the two files were mosaicked into one DEM, we masked it with a U.S. national boundary to separate terrain from background pixels and data in Canada and Mexico. To increase tonal contrast in

smooth topography and diminish it in areas of high relief, we reduced skewness of the elevation frequency distribution, by transforming all 12 million elevations to their logarithms (see also USGS, 1986).

The SUNSHADE* routine (ESL, 1983; Scholz et al., 1987), which computed the angles of terrain slope and aspect (azimuth) and assigned to them brightness values and the corresponding 256 shades of grey, employs a modified Lambertian, or diffuse-scattering, photometric function. The following parameters were used: vertical exaggeration 2X, sun azimuth 300° , sun elevation 25° , image intensity 1.2 units, image ambience 0.7 units. We examined the histogram of resulting brightness values and redistributed the range to truncate its tails, and thus further reduce tonal imbalance between steep and gentle terrain. From an output tape of stretched brightnesses, we made six negatives on an Optronics* C-4500 Color Scanner and Film Recorder*. Lastly, we enlarged the negatives to prints of the same scale as the standard geologic map of the U.S. and mosaicked them to yield the 1:2,500,000-scale wall map.

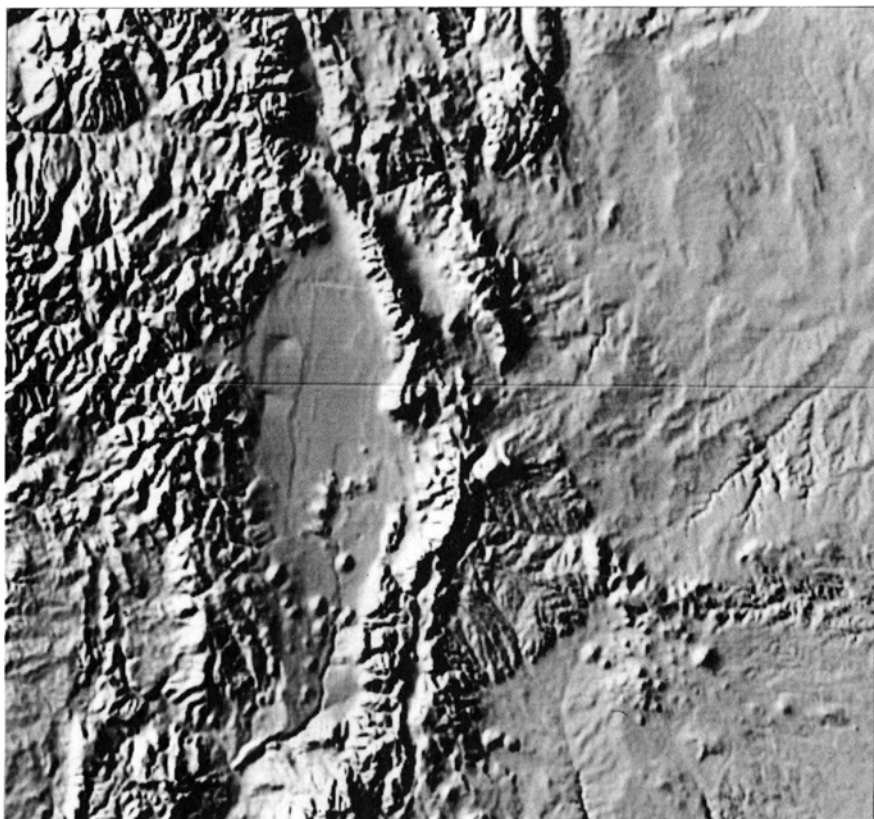


Figure 6. Detail from the 1:2,500,000-scale U.S. shaded-relief map, centered on the San Luis Valley and Sangre de Cristo Range, Colorado, including parts of Southern Rocky Mountains (Province 16) and the Raton (13g) and Colorado Piedmont (13f) sections of the Great Plains province. Horizontal line is mosaic seam. Area shown is 192 km across.

The resulting image (Figures 2, 6) is the first one-sheet graphic of U.S. landforms larger than Raisz's (1939) hand-drawn map. Fidelity and detail are far greater than that evident on Raisz's and other synoptic portrayals of terrain by pictorial relief, airbrush, or dark-plate (Harrison, 1969). The new map shows regional geomorphic and tectonic features not readily viewed by other means. It complements Hammond's (1964) map depicting numerical classes of land-surface form, and will aid in analyzing the Fenneman map units and adjusting their boundaries. The map also has revealed the worst errors in the data, which are being corrected before its publication at 1:2,500,000.

The new image can be improved, by employing local operators within SUNSHADE (Brassel, 1974), by experimenting with other advanced techniques of image processing (Whitted, 1982), and by adding color (Lamb et al., 1987). Maps at several sun angle and azimuth settings (e.g., Moore & Simpson, 1982) will be required to accentuate terrain features that follow different trends in the U.S. landscape.

CONCLUSIONS

Statistics of elevation and slope for 82 physiographic sections provide the first quantitative basis on which to summarize and compare topographic properties of the conterminous U.S. from so many map units. The results, derived wholly by image-processing a large file of terrain heights, both reveal that the Fenneman subdivisions vary widely in topographic homogeneity and yield criteria for refining the taxonomy. The accompanying wall map of U.S. topography in shaded relief, also from digital image-processing, is the best single-sheet graphic yet produced of the nation's landforms. Its accuracy and detail are unprecedented. Both the map and the geometric signatures extracted from elevation derivatives are new tools for addressing problems of regional extent in geology and geography.

ACKNOWLEDGMENTS

The Photo-Imagery Unit of USGS' Western Mapping Center made the map prints, and R.O. Castle and R.K. Mark clarified the presentation. We thank R.E. Slye and K. Weinstock for programming assistance.

REFERENCES

- Arvidson, R.E., Guinness, E.A., Strebeck, J.W., Davies, G.F., & Schulz, K.J., 1982, Image-Processing Applied to Gravity and Topography Data Covering the Continental U.S.: EOS, Vol. 63, No. 18, pp. 257, 261-265.
- Batson, R.M., Edwards, K., & Eliason, E.M., 1975, Computer-Generated Shaded-Relief Images: Jour. Research USGS, Vol. 3, pp. 401-408.
- Brassel, K., 1974, A Model for Automatic Hill-Shading: American Cartographer, Vol. 1, pp. 15-27.
- Electromagnetic Systems Laboratory (ESL), Inc., 1983, IDIMS Users' Guide, Vol. 1, Sunnyvale, California.
- Fenneman, N.M., 1931 & 1938, Physiography of Western United States & Physiography of Eastern United States: New York, McGraw-Hill.
- Godson, R.H., 1981, Digital Terrain Map of the United States, U.S. Geol. Survey Miscellaneous Investigations Map I-1318, 1:7,500,000.

- Hammond, E.H., 1964, Classes of Land Surface Form in the Forty Eight States, U.S.A.: Annals Assoc. Amer. Geographers, Vol. 54, No. 1, Map Suppl. No. 4, 1:5,000,000 (also article, pp. 11-19 in same volume).
- Harrison, R.E., 1969, Shaded Relief: National Atlas of the United States, Sheet No. 56, 1:7,500,000.
- Horn, B.K.P., 1981, Hill Shading and the Reflectance Map: Proc. IEEE, Vol. 69, No. 1, pp. 14-47.
- Lamb, A.D., Malan, O.G., & Merry, C.L., 1987, Application of Image Processing Techniques to Digital Elevation Models of Southern Africa: So. African Journal of Science, Vol. 83, pp. 43-47 and cover (color).
- Mark, D. M., 1975, Geomorphometric Parameters: A Review and Evaluation: Geografiska Annaler, Vol. 3-4, Ser. A, pp. 165-177.
- Mays, R.R., 1966, Production of Numerical Maps: Numerical Topographic Data and Other New Map Products, Proceedings, 3-5 May 1966 GIMRADA Conference, Fort Belvoir, Virginia, pp. 38-41K.
- Moore, J.G., & Mark, R.K., 1986, World Slope Map: EOS, Vol.67, No. 48, pp. 1353 & 1360-1362.
- Moore, R.F., & Simpson, C.J., 1982, Computer Manipulation of a Digital Terrain Model (DTM) of Australia: BMR Jour. of Australian Geology & Geophysics, Vol. 7, pp. 63-67.
- Noma, A.A., 1966, Current AMS Computer Processing of Numerical Topo Data: Numerical Topographic Data and Other New Map Products, Proc. 3-5 May 1966 GIMRADA Conference, Fort Belvoir, Virginia, pp. 42-48.
- Pike, R.J., 1988, The Geometric Signature: Quantifying Landslide-Terrain Types from Digital Elevation Models: Mathematical Geology, Vol. 20, pp. 491-511.
- Pike, R.J., & Acevedo, W., 1988, Image-Processed Maps of Southern New England Topography: Geol. Soc. America Abstracts with Programs, Vol. 20, No. 1 (Northeast Section annual meeting, Portland ME), p. 62.
- Raisz, E., 1939, Landforms of the United States, Cambridge, Mass., one sheet, ca. 1:4,500,000 (sixth revised edition, 1957).
- Scholz, D.K., Doescher, S.W., & Hoover, R.A., 1987, Automated Generation of Shaded Relief in Aeronautical Charts: Tech. Papers 1987 ASPRS-ACSM Annual Convention, Vol. 4 (Cartography), pp. 212-219.
- U.S. Geological Survey, 1986, Production of Shaded-Relief Products: Geological Survey Yearbook 1986, pp. 37-38.
- Whitted, T., 1982, Some Recent Advances in Computer Graphics: Science, Vol. 215, pp. 767-774.
- Wood, W.F., Soderberg, P.G., & Pike, R.J., 1962, A Preliminary Model of Most-Severe Contour Flying. Aircraft-Environmental Research Study Report AE-4, U.S. Army Quartermaster Research & Engineering Command, Natick, Massachusetts, 33 pp.
- Yoeli, P., 1967, The Mechanisation of Analytical Hill Shading: Cartographic Journal, Vol. 4, pp. 82-88.

A Full Function GIS Editor

William H. Moreland

Environmental Systems Research Institute
380 New York Street
Redlands, California 92373

ABSTRACT

With the introduction of engineering workstations, GIS graphic editors now have the means to provide the user a single interface combining the capabilities of high speed graphics along with a fully functional two-way link to a relational database. A GIS editor must perform more than graphic edits, it must also maintain the integrity of a GIS database while editing both coordinates and attributes. The GIS editor needs to consider both the coordinates and attributes of each feature as a single entity; and to operate upon each entity with fast graphics operations as well as allowing full attribute editing capability. This paper will outline the requirements and specification of such an editor.

Introduction

The most common type of editors edit a single data type or file; but since a GIS database is not a single data type, but in fact a collection of different data types which together form a database of both spatial and non-spatial data (figure 1). A person wishing to edit a GIS database does not want to edit any given part of the database as a single data type, but to alter the database as a whole. Therefore, a GIS editor must consider the GIS database as a whole and allow the user to alter it similarly. Most editors to date have performed well on either of the two main parts of a GIS database; the coordinates (spatial) or the attributes (non-spatial), but not both.

A Full Function GIS Editor

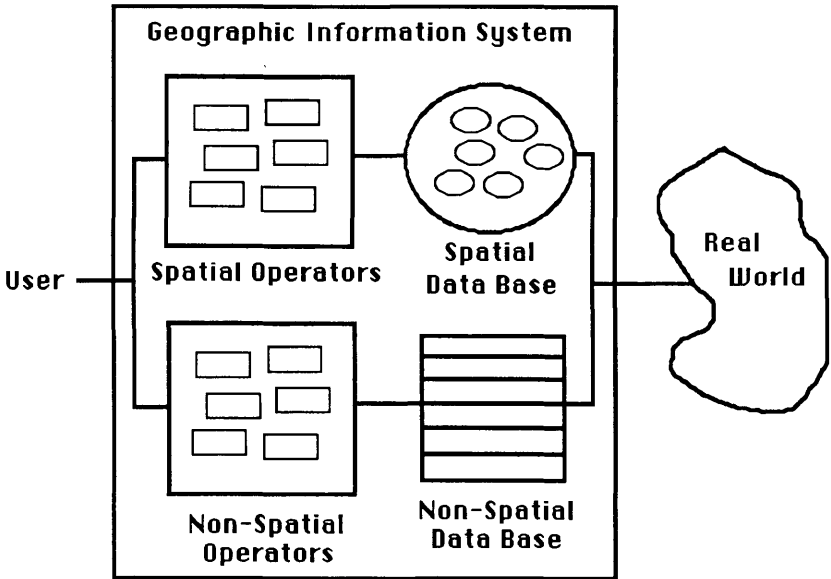


Figure 1.

There is a third component of a GIS that most editors seem to conveniently forget: topology. Topology is the glue that holds the GIS database together. The end result of most GIS editors is the edited versions of the two main components, that still need to be glued together. What is needed is an editor that allows fast efficient update of both of the spatial and non-spatial components while maintaining the topology. A fully functional GIS editor must produce a fully functional GIS database. What this paper will describe is how ARCEDIT has evolved over the years in response to the growing requirements for GIS editor, and our future plans for it.

A Full Function GIS Editor

A GIS editor needs to allow the user to alter and view any aspect of the low level components of the database. For ARC/INFO coverages these components are: Arcs, Nodes, Points, Annotation, and Tics (figure 2). The attributes (if any) of each of these components are linked with their counterparts to form a single entity capable of being edited (figure 3). This is ARCEDIT's basic purpose, since the coordinates and the attributes of each component are treated as a whole the editor not the user worries about the link between them and how that changes when either is altered. All the user cares about is that he has either moved the location or changed the value of an attribute.

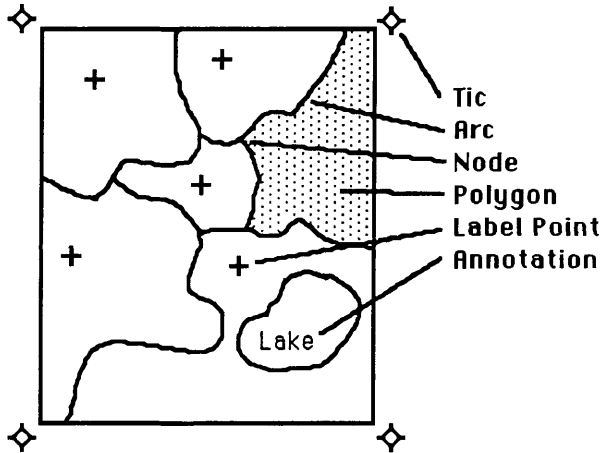


Figure 2 (coverage).

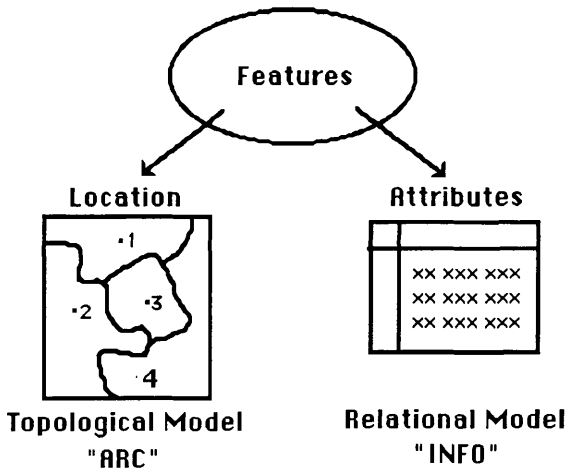


Figure 3.

A Full Function GIS Editor

As would be expected in a graphic editor, ARCEDIT allows any feature to be moved, copied, rotated, deleted, added, etc; as well as allowing any of the attributes to be calculated, assigned, etc. ARCEDIT satisfies the requirements of a GIS editor, by providing the user with environments to perform as much of the functions of the CLEAN and BUILD process as possible while maintaining a good response time.

CLEAN and BUILD are the topology builders within ARC/INFO. CLEAN takes as input the arcs and/or points (labels) finds all intersections between segments, and processes them into polygon or line coverages. BUILD and CLEAN perform basically the same operation, but BUILD skips the intersection phase in order to save time.

ARCEDIT has a number of editing environments that control how coordinates are handled while editing. These five environments: nodesnap, arcsnap, snapping, intersectarcs, and attribute only. These environments are responsible for ensuring that the nodes (end points of arcs) snap to their neighbors if within tolerance, resolving intersections within the arcs, snapping any component with any other from any other database, resolving both undershoots (arcs that are suppose to intersect but fall short) and overshoots (arcs that are suppose to meet exactly with another), and allowing non-spatial edits to not destroy the topology.

The snapping environments work in three ways. First all nodes are always snapped to each other when ever an arc is updated. This is to ensure that all arcs that are suppose to meet at a single node do in fact share the same coordinates for the common node. The second environment allows any feature being altered to be snapped to any other feature either within the same or any other coverage. This allow different feature types that are suppose to overlap and meet, to in fact do so and to have the same coordinates. The final snapping environment is where under and over shoots are resolved upon the adding of new arcs. This is used to ensure that those arcs that are suppose to end exactly at another do in fact do so and that where it does meet the other arc that, that arc is split and a node is generated. This is the only environment that is active only upon the adding of new arcs, and not during the entire edit session.

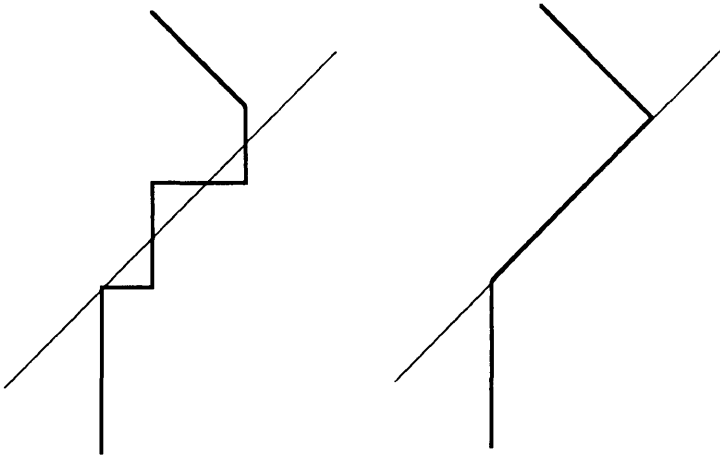
The intersection environment performs much of the preprocessing that goes on within CLEAN by ensuring that all the possible intersections between arcs have already been resolved and therefore it is not necessary to use CLEAN, but only reestablish topology by only using BUILD. Each arc as it is edited is checked against the region of the database that it overlaps for intersection with any existing arc, then the arc is checked against itself. Each new segment of the edited arc will each possess its own copy of the initial attributes.

The last environment is not really an environment at all, but in reality a internal flag that keeps tract of any spatial edits. If at the end of the edit session no spatial edits were performed, then only the non-spatial side of the database is altered and the topology of the spatial side remains intact. This allows Users to use the spatial capabilities of a GIS editor to select the components for update and to edit the attributes directly. This is where, once the database is built, most of the edits will take place. The user views the database as a whole and should be allowed to edit it as one; and not be forced to use one editor for spatial updates and another for non-spatial.

A Full Function GIS Editor

All of these environments are checked only when coordinates of any feature are updated and are order dependent. In the case of arcs, after each arc is edited, but before it is added back into the coverage, it is acted upon in the following order:

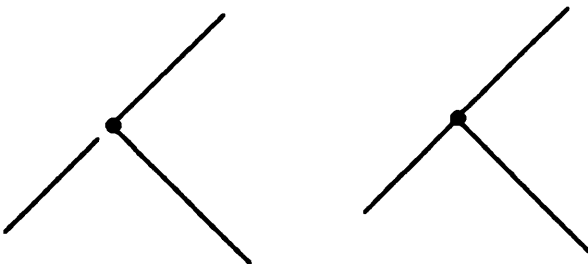
- 1) **SNAPPING** Check arc for snapping to any other feature of this or another coverage. This snapping environment is valid for all features, where as the rest are only valid for arcs.



SNAPPING

- 2) Check the nodes of the arc for snapping onto each other. In this case the last (to) is snapped onto the first (from). This ensures that islands (polygons represented by a single arc) are closed. This is not an environment, but does use the snap tolerance set by NODESNAP.

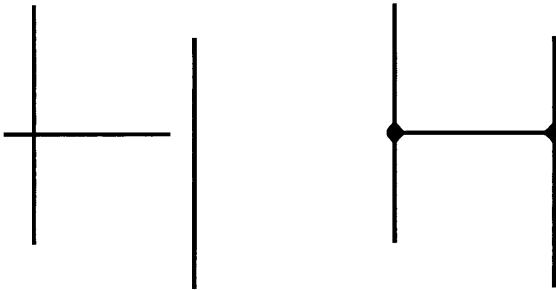
- 3) **NODESNAP** Check the nodes of the arc for snapping onto all of the existing nodes within the region of the coverage defined by the confining box of the altered arc. This environment is for arcs only.



NODESNAP

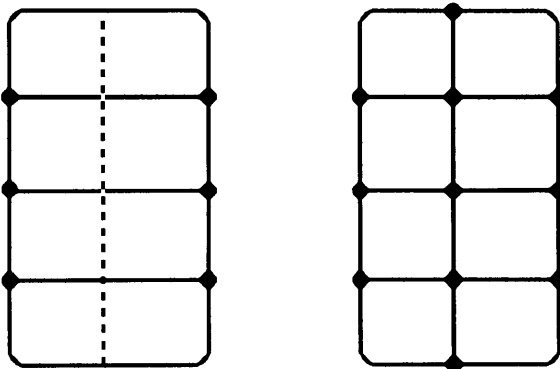
A Full Function GIS Editor

4) **ARCSNAP** Check for under or over shoots. This environment is for arcs only.



ARCSNAP

5) **INTERSECTARCS** Check for intersections with either existing arcs or itself. This environment is for arcs only.



INTERSECTARCS

6) Recalculate the length of the arc and add to the coverage. This is not an environment, but is performed for every arc.

A Full Function GIS Editor

The non-spatial side of the database, as discussed earlier, should be fully accessible and changeable from a GIS editor. The user needs to be able to view and change any attribute of any feature, and to be able to establish selection sets by either attribute equation or by spatial selection. In other words, the user should be able to spatially alter any set of features derived by a attribute equation; and to be able to alter the attributes of any set of features derived by a spatial select. Since ARCEDIT considers the GIS database as a single entity in principle as well as in practice, it allows the user to select, view, and update any aspect of the database.

As with all editors, there needs to be a way of recovering from making mistakes. All updates to a database must not be irreversible, until such time as users save their updates. The philosophy of ARCEDIT is to only access the database in a Read-Only mode placing all updates into local scratch files. It is this philosophy that allows ARCEDIT to be able at any time, OOPS any number of updates all the way back to the beginning of the edit session, and to support recovery of updates in the case of the computer system going down.

The last thing all GIS editors needs to do is to maintain the topology. The topology is a part of the GIS database and therefore, should be accessible and maintainable by the editor. The above described editing environments go a long way toward accomplishing many of the low level steps required for maintaining topology interactively, but the final step of rebuilding the arc segments into polygons is not taken. We plan to take this final step in the next revision of ARCEDIT (ARC/INFO version 6.0). It is the introduction of engineering workstations, that make it possible to deliver topology on the fly while maintaining good interactive response time.

Conclusion

Basically a full function GIS should be able to work with the GIS database in its entirety and to present at all times an intact and fully functional GIS database to the user. This implies that upon completing the edit session the end result, if given a fully functional GIS database at the onset, is a fully functional GIS database. With the introduction of engineering workstations, and the steps already incorporated within ARCEDIT, there is enough compute power to ensure good response time while maintaining topology on the fly.

A STUDY OF SPATIAL DATA MANAGEMENT
AND
ANALYSIS SYSTEMS

Clyde Christopher
Jackson State University
Jackson, MS

Richard Galle
Stennis Space Center
NSTL, MS

ABSTRACT

The Earth Resources Laboratory of NASA's Stennis Space Center is a center of space related technology for earth observations. It has assumed the task, in a joint effort with Jackson State University, to reach out to the science community and acquire information pertaining to characteristics of spatially oriented data processing.

INTRODUCTION

During the past 15 years much computer software has been developed for handling spatial data. A number of centers have been set up for processing spatial data, all with different configurations. Some studies have been conducted to collect and disseminate data on software in this area. Scientists at the Science and Technology Laboratory were interested in knowing the software and hardware characteristics in such centers. Additionally, they wanted to know what centers those who are actively engaged in spatial data processing are communicating with. This information is to be shared with scientists in those centers so that all will be fully aware of the state-of-the-art software and hardware for spatial data processing.

METHODOLOGY

The desired information was collected by means of a survey. The survey instrument was designed by Christopher and Galle with approval from proper authorities. Part I asked for background information including names of administrators, primary mission, list of principal application areas, and a list of representative projects currently supported at that facility.

Part II requested data processing methodology. First, it asked for a listing of hardware devices devoted to spatial data processing including the host computer, display devices, digitizers, disk drives, tape drives, plotters, etc. Next, it asked for a list of software packages currently supported by the facility, including compilers, GIS systems, statistical packages, data base management systems, etc. Finally, it asked with

what other facility they were linked and what communication package is being used.

Part III is a request for data characteristics. The survey asks what types of data are used (Landsat, TM, AVHRR, Soils Maps, etc.), what are the data sources, how is it input for processing, what is its internal format (Raster, Vector, etc.), and what CAD related capabilities are available for editing the data sets.

Part IV asked for data analysis characteristics. It specifically asked what types of data processing (statistical, analytical, expert system decision making, polygon declaration, model structuring) supports the applications at that installation. Also, is there merging or overlaying of data and what steps are taken in the processing life cycle.

Part V, the concluding section, asked for representative research which has been undertaken at the facility in the area of spatial data processing and, also, the research objectives which are needed.

The survey was first mailed under a cover letter dated July 10, 1987. A second copy was mailed to those facilities that had not responded by September 8, 1987. The survey was mailed to 349 different installations. Completed forms were received from 115 installations. A survey instrument worded so that all answers could be given by checking the correct item probably would have yielded a higher number of responses. However, that type of instrument would not yield as much detailed information as the one used.

The data will be used to build a data base with appropriate query language so that local users will have access to it. Hopefully, problems in communication will be solved providing access to this data on a nationwide basis.

RESULTS

Background Information

The survey was sent to basically three types of organizations: 1) government agencies, 2) private industry, and 3) educational institutions. In addition to supplying names of key contact personnel at the installation this section of the survey provided us with a statement of the primary mission of the installation and a list of ongoing projects. The government and industrial organizations said they were involved in research in remote sensing and in analyzing terrain and giving technical advice on use of land and water resources. The educational institutions are involved in training students in the use of remote sensing as well as research in that area.

No two installations listed the same project. Typical projects were: "Land use of 13 county region of Tennessee", "Training in forestry and wetlands remote sensing techniques", and "produce 7½ foot wetland maps of the United States". Some indicated that project titles are not available to the public.

The principal application area is an important response in this section. These ranged from "land use classification" to "Geological engineering". However, the most frequent responses were "remote sensing", "image processing", "GIS", or some combination of these.

Data Processing Methodology

Computer systems used to process spatial data range from microcomputers to mainframes. The most frequently listed microcomputers are the IBM PC-AT and the APPLE II PC. In the minicomputer class the VAX 11/780 is most used. In the largest class an Amdahl V7/V8 was listed by one installation. The prime 9650 was listed by several installations.

Table 1. Host Computer(s)

<u>Model</u>	<u>Number of Installations</u>
Microvax II	15
Apple IIc	2
Amdahl (5860/V7/V8)	3
Harris (1000,500,800)	3
VAX 98200,11/730,11/750, 11/780,11/785)	36
Hewlett-Packard 9000	2
IBM PC (ATXT)	12
SUN	3
PDP 11/70	3
Concurrent (Perking Elmer)	5
Prime (400, 550, 750, 465, 9755, 9650, 2655, 2275, 6350, 250, 9955)	
Data General	4
Gould 32/27	5
IBM (3013, 3081, 3084, 4365, 4381)	11
Zenith 248	3
MASSCOMP 5600	5
AT&T (3B5, 3B2)	3

Table 2. Operating Systems

<u>Software</u>	<u>Number of Installations</u>
RIPS	1
VMS	39
VOS 6.1	2
UNIX	11
PRIMOS	30
MVS	7
RT-11	3
MS-DOS	6
MPX	4

RSX-11M	2
OS-32	5
VM/CMS	3
VORTEX	2
AOS	2
ULTRIX	1

Table 3. Digitizers

<u>Model</u>	<u>Number of Installations</u>
GTCO	15
Hitachi	4
CALCOMP 9100	32
Altex	23
Geographics	5
Summagraphics	6
Perkin Elmer (concurrent)	1
Talos	5
NUMONICS	3
Integraph	2
Tektronix	7

Table 4. Display Devices

<u>Model</u>	<u>Number of Installations</u>
Tektronix	58
Comtal	11
De Anza	3
ADAGE	2
Visual 500	8
CALCOMP	2
Lexidata	6
Hewlett Packard	3
Inter Graph	2
ERDAS	3
Hitachi	3

The type of compilers available depends upon the model of computer and the memory size. Some installations list Fortran, only. Pascal and C are also frequently listed. Many installations listed all three of the above languages, but Fortran was listed most frequently. Other languages used included Lisp, Prolog, BASIC, SCAN, COBOL, Modula2, Ada, and PL/1.

Among the image processing and GIS systems listed, ELAS and ARC/INFO were most frequent. Others in the order of occurrence were RIPS, ORSER, GRASS, ERDAS, MOSS, SYNERCOM, and ODYSSEY.

Table 5. Image Processing Software

<u>Model</u>	<u>Number of Installations</u>
ELAS	30
RIPS	6
COS	3
ARC/INFO	45
GRASS	6
ERDAS	22
MOSS	22

SYNERCOM	2
ORSO	2
ATLAS	2
Inter Graph	2
EROS	2

Some installations indicated that their processing involves statistical analyses and they named the statistical package that is available on their system. SAS, SPSS, GLIM, BMDP, MINITAB, OSIRIS, NAG, VECTOR, MATH 77, SSPLIB, IMSL and Microstat are among the packages named.

Database management systems are as varied as statistical packages. However, some who responded did not name a DBMS. One installation stated that it has an inhouse information storage and retrieval system. The usual response was one or more of the familiar systems such as Datatrieve, INFO, PC INF, RBASE, Dbase III, ORACLE, BASELINE, CDOS, SMARTSTAR, SPIRES, or INGRES.

Table 6. Data Base Management Systems

<u>Software Package</u>	<u>Number of Installations</u>
DBASE II or III	20
Datatrieve	5
INFO	36
SMARTSTAR	1
ORACLE	7
BASELINE	1
RBASE	5
Prime-INFORMATION	4
SPIRES	1
OPS-83	1
CDOS	1
INGRES	4

Some large installations at government sponsored organizations indicated that they are connected with state or regional offices through some type of communication link. In general, the educational institutions have local networks. This was the case with several other installations as well, but there were those who indicated that they have no networking.

Data Characteristics

Landsat, TM and MSS data is the most used data in the installations that responded. However, a great variety of data types are used. AVHRR, Soils Maps, SPOT, and GOES were frequently named in the survey. Most installations named several types. Some other types listed were land use maps, USGS topographic maps, ACZCS, census, transportation networks, streams and rivers, watershed and aerial photos. The primary sources of data are EROS, NASA, USFS, USGS, SPOT Image Corporation, and virtually all map sources.

Table 7. Data Types

<u>Data Type</u>	<u>Number of Installations</u>
LANDSAT (MSS AND TM)	72
AVHRR	24
Soils Maps	41
Land Use Maps	12
Census	10
SPOT	21
GOES	6
SIR-B	2
USGS Topographic Maps	18

As a rule digitizers and magnetic tapes are used to input data for processing. Floppy disks were frequently listed as input media. Video cameras, optical disks, and keyboard were each listed by at least two installations.

Table 8. Data Input Scheme

<u>Input Media</u>	<u>Number of Installations</u>
Digitizer	84
Magnetic Tape	86
Magnetic Disk	21
Optical Disk	3
Keyboard	9
Scanning	6
Digital Camera	6

The internal format for data was given as raster or vector or both. CAD related capabilities at various installations include scroll, zoom, draw, density slice, classification, cut, paste, paint.

Table 9. Input Format

<u>Characteristic</u>	<u>Number of Installations</u>
Raster	86
Vector	72
Gridded	4

Data Analysis Characteristics

In answer to the question on analysis of data during processing, we found that a majority of installations do a statistical analysis. Several Installations use the data in decision making for expert systems. A few installations are doing polygon declarations. A small number indicated that they are merging data sets and at least one installation is overlaying data. Some installations were cooperative in sharing with us the steps in the processing life cycle, not an easy task.

CONCLUSION

In this section some installations listed some research efforts that are ongoing in special areas. A non-conclusive list of these follows:

- 1) Incorporation of the SOI-5 soil interpretation records into ARC/INFO
- 2) Incorporate spatial data (GIS) into a decision support system
- 3) GRASS G&D functions
- 4) Link between S and GRASS
- 5) Hydrologic modeling, flood damage analysis
- 6) Image texture recognition
- 7) Geometric rectification, image animation
- 8) Map display techniques
- 9) Digital image processing-both spectral and spatial analysis

(Notes. The researchers acknowledge assistance in this project from the following: Pauline Frances, System Analyst; Jeanette Lewis, Laboratory Coordinator; Andrew Ward, Kimberly Wallace, Rickey Myers, Jody B. Hasten, and Shelton James, students; Jackson State University.)

Sliding Tolerance 3-D Point Reduction for Globograms

Steven Prashker
Cartographic Research Unit
Department of Geography
Carleton University
Ottawa, Ontario, Canada K1S 5B6

ABSTRACT

Vector maps that are plotted using a globogram type of projection may have linework that becomes crowded when approaching the map's horizon line on the hardcopy plot. To reduce this linework clutter on the globogram map, the map can be passed through a point reduction algorithm to thin the points. Point reduction algorithms, whether local or global in nature, typically are two-dimensional geometrical processes, that operate equally on all of the map's digital lines. In the globogram case, this can result in too much point reduction in the central, foreground areas of the map and too little point reduction at the extreme areas near the horizon line, where the oblique 'viewing angle' increases the density of points.

To solve the above problem, this paper will propose and demonstrate an implementation of a three dimensional method of point reduction using a modified local point reduction algorithm. The proposed methodology and algorithm will utilize a variable or sliding tolerance criteria for the point reduction based on a selected point's proximity to the horizon line of the globogram map.

BACKGROUND

In digital mapping, point reduction algorithms are applied to maps to remove excess or unnecessary points while maintaining the basic caricature or shape of the lines. The application of these routines may be required due to the reduction in scale of a map. Added benefits of point reduction include reduced plotting time, reduced storage space of the coordinate pairs, faster vector processing and faster vector to raster conversion (McMaster, 1987).

Over the last twenty years, several algorithms have been developed to perform automated point reduction on digital maps. A review of automated line generalization and point reduction by McMaster (1987) contained discussions of several point reduction algorithms. These algorithms use two dimensional geometrical techniques to remove points from two dimensional maps. The current research sought an improved technique for point reduction for the particular problems associated with the globogram projection.



Figure 1. ORIGINAL MAP - 3445 POINTS



Figure 2. GLOBOGRAM MAP - 3445 POINTS

THE GLOBOGRAM PROBLEM

The globogram is a two-dimensional representation of the three-dimensional map model. Viewed orthogonally, a globogram map gives the appearance of being projected on a sphere, since it is defined in three dimensions and plotted in two dimensions. A map defined in spherical coordinates (R, θ , ϕ) or cartesian coordinates (x,y,z) can be rotated in any of the x, y or z directions, and subsequently plotted using the x and y coordinates on a globogram projection. The problem of line clutter occurs when the z coordinates (in cartesian space, the x,y plane is the map sheet plane) of the map boundaries approach zero, i.e. the map boundaries approach the horizon line. Figure 1 shows the original map of Canada, using the Lambert Conformal projection. Figure 2 shows the globogram of the original map (original map digitally placed on a 400 mm radius sphere, then rotated 58 degrees about the Y axis). Both maps were plotted from the same source basefile containing 3445 coordinates. Due to the nature of the globogram projection, northern portions of the Canadian coastline become crowded and the coordinates become more densely clustered. Less detail is required in these northern areas, and in the general case, less detail is required as map areas approach the horizon of the globogram projection.

TWO-DIMENSIONAL POINT REDUCTION METHODS

Two sequential methods of two-dimensional point reduction may be employed in an attempt to alleviate the point density of the lines. These methods are:

- A. performing a two dimensional point reduction on the original map's x,y coordinates, and then projecting the map onto a globogram,
- B. projecting the original map onto a globogram, and then performing a two dimensional point reduction on the globogram.

Any of the local or global point reduction algorithms are suitable for these two methods; however, these algorithms follow similar methodologies: point reduction is performed equally on all the lines that comprise the map due to a controlling set of fixed tolerance criteria. These criteria may be a tolerance distance, a perpendicular distance or deviation, a field of view angle or deviation, or any other combination of threshold values. The common factor of these algorithms is that this threshold tolerance value is a fixed quantity, usually supplied by the user. Applying these algorithms to two dimensional maps generally produces good results everywhere on the point reduced map, depending on the algorithm used and the selected threshold tolerances. However, the globogram is a special case that requires varying degrees of point reduction that is not provided with traditional two-dimensional methods, as seen in the following examples.



Figure 3. 50% REDUCTION - 1720 POINTS



Figure 4. GLOBOGRAM OF 50% REDUCTION - 1720 POINTS



Figure 5. GLOBOGRAM MAP - 3445 POINTS



Figure 6. 50% REDUCTION OF GLOBOGRAM - 1721 POINTS

TWO-DIMENSIONAL POINT REDUCTION FOR GLOBOGRAMS

A comparison was made between three two-dimensional point reduction algorithms:

1. the proximity method
2. the Jenks modified angular method
3. the perpendicular/proximity method

using two reduction factors, 32.5% and 50%, for the two above sequential methods. After examining the output, it was determined that the perpendicular/proximity method produced the best results in this particular case, and that the 50% reduction demonstrated the most dramatic effects of the algorithm.

Method 'A' Globogram Point Reduction

Figure 3 is the output of the perpendicular/proximity algorithm (50% reduction) as applied to the original map, while Figure 4 shows the globogram of the 50% reduction map. There is a noticeable difference in the point density of the globogram of the point reduced map when compared with Figure 2. However, this reduction in the point density of the lines occurs everywhere on the globogram map, and crowding is still evident in the northern sections. Also, undesirable thinning of the lines has occurred in the southern portions of the map.

Method 'B' Globogram Point Reduction

Figure 5 is the globogram of the original map while Figure 6 shows the results of the perpendicular/proximity algorithm (50% reduction) as applied to the globogram of the original map. Again, there is a noticeable difference in the point density of the reduced globogram when compared with Figure 2. However, as in method A, the reduction in the point density occurs everywhere on the globogram map, and the point crowding in the northern sections has not sufficiently improved, while excessive point reduction has occurred in the southern sections.

THE SLIDING TOLERANCE POINT REDUCTION

The globogram case requires the application of a selective point reduction algorithm that removes more points at the critical areas near the horizon line, and removes relatively fewer points in the less oblique foreground areas.

Since the critical areas of a globogram projection are lines that approach the horizon line, where the z coordinates of the vertices approach zero, point reduction solutions were sought which made effective use of this z coordinate. The z coordinates would be used to evaluate the globogram line's proximity to the globogram horizon line. As a line approaches the horizon, convergence of its vertices increases due to the

oblique viewing angle. To increase the map's legibility, the elimination of a greater number of points in the horizon areas is required.

Global point reduction algorithms, such as the Douglas-Peucker algorithm (Douglas and Peucker, 1973) are not suitable since they operate on a complete line. The group of local processing point reduction algorithms, where a point is tested for redundancy relative to its immediate neighbours, was determined to be appropriate and easily adaptable for this application. Instead of being fixed, the threshold value for each algorithm was allowed to vary from a user-defined minimum value to a user-defined maximum value, following a function that was based on the z coordinate of the test vertex.

Three sliding functions, that varied the range of the tolerance values were tested:

Theoretical Function	Actual Function Used
1. $(R-Z)/R * \text{Tolerance}$	$(R-Z)/R * \text{Tolerance}$
2. $0/90 * \text{Tolerance}$	$[\text{ARCCOS}(Z/R)]/90 * \text{Tolerance}$
3. $\text{SIN}(0) * \text{Tolerance}$	$\text{SQRT}(1-(Z/R)^2) * \text{Tolerance}$

where Tolerance was a user-defined tolerance range. A typical formula for computing the actual value of the tolerance value, using function 1 is:

$$\text{TOLVAL} = \text{MINTOL} + \underbrace{\left[\frac{(R-Z)}{R} * (\text{MAXTOL} - \text{MINTOL}) \right]}_{\substack{\text{sliding} \\ \text{function}}} \underbrace{\hspace{10em}}_{\substack{\text{tolerance} \\ \text{range}}}$$

where TOLVAL = computed tolerance value for the test vertex
MINTOL = user-defined minimum tolerance value
MAXTOL = user-defined maximum tolerance value
R = radius of the globogram sphere
Z = z coordinate of the test vertex

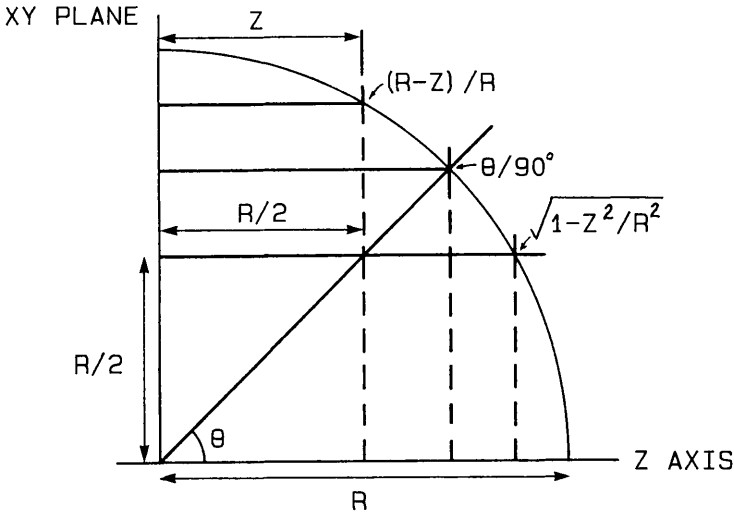


Figure 10. GEOMETRY OF SLIDING FUNCTIONS



Figure 7. GLOBOGRAM OF 50% REDUCTION - 1720 POINTS



Figure 8. 50% REDUCTION OF GLOBOGRAM - 1721 POINTS



Figure 9. 50% GLOBOGRAM 3-D REDUCTION - 1723 POINTS

Each of these functions has its own unique characteristics, being linear, transcendental and quadratic respectively, resulting in different responses or sensitivities on the tolerance values. All the sliding functions range in value from 0 to 1 for Z ranging from R to 0. This results in minimum tolerance values and point reductions when $z=R$ and maximum tolerance values and point reductions when $z=0$. Figure 10 gives a representation of the geometry involved in each of the sliding functions, and shows the position of a point on the sphere where the computed value of TOLVAL is halfway between MINTOL and MAXTOL. The choice of sliding function is not unique, and the final results would clearly depend upon the amount of curvature and rotation in the globogram.

RESULTS

The perpendicular/proximity point reduction algorithm, using the linear function $(R-Z)/R$ to vary the tolerance values, provided the best results, at a 50% reduction in total points, for the Canada example. An illustration of this three-dimensional technique can be seen in Figures 7 through 9. For comparison purposes, Figures 7 and 8 repeat the previous Figures 4 and 6 respectively. Figure 9 shows the results of the adapted algorithm. A significant reduction in the number of points has occurred in the northern sections of the map, near the horizon, whereas much less reduction has occurred in the areas near the foreground. When comparing the three techniques, it is evident that the sliding tolerance method gives significantly better results in areas requiring the most point reduction, leaving the foreground areas relatively intact. Because of the utilization of the sliding tolerance criteria, this approach is superior to the other two methods.

CONCLUSION

The usage of conventional two-dimensional algorithms for enhancing the display of globograms can be significantly improved by incorporating the 3-D sliding tolerance methodology. The utilization of the z dimension as a controlling parameter in determining the tolerance for conventional two-dimensional point reduction algorithms allows for varying degrees of point reduction depending on the relative position of the map on the globogram sphere. This adaptive technique can be successfully applied to any of the 'local' two-dimensional point reduction algorithms, resulting in a more satisfactory presentation of the globogram map.

Future directions and research include evaluations of characteristics of different sliding functions as adapted to various local two-dimensional point reduction algorithms, the display of map projection error as a function of the line detail of the map, and the analysis of the perception of visual density of globograms

ACKNOWLEDGEMENTS

The author wishes to thank David Broscoe and Christine Earl for their much appreciated contributions and suggestions.

REFERENCES

Douglas, D.H. and Peucker, T.K. 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature: **The Canadian Cartographer**, Volume 10, Number 2, pp. 112-122.

McMaster, R.B. 1987. Automated Line Generalization: **Cartographica**, Volume 24, Number 2, pp. 74-111.

Prashker, S. 1988. Sliding Tolerance 3-D Point Reduction Geometry Technical Report: **Cartographic Research Unit Technical/Research Series**, Number TT-03.

A Reactive Data Structure for Geographic Information Systems

Peter van Oosterom
TNO Physics and Electronics Laboratory,
P.O. Box 96864, 2509 JG The Hague, The Netherlands
and
Department of Computer Science, University of Leiden,
P.O. Box 9512, 2300 RA Leiden, The Netherlands
(Email: OOSTEROM@HLERUL5.BITNET).

January 10, 1989

Abstract

We introduce a *Reactive Data Structure*, that is a *spatial* data structure with *detail levels*. The two properties, spatial organization and detail levels, are the basis for a Geographic Information System with a multi-scale database. A reactive data structure is a novel type of data structure carting to S It is presented here as a modification of the binary space partitioning tree that includes detail levels. This tree is one of the few spatial data structures that do not organize the space in a rectangular manner. An application of the reactive data structure in thematic mapping is given.

1 Introduction

In the past few years there has been a growing interest in *Geographic Information Systems* (GISs). There are many applications that use GIS technology, among them: Automated Mapping / Facility Management (AM/FM); Command, Control and Communication Systems (C³S); War Gaming; and Car or Ship Navigation Systems. A major advantage of a GIS over the paper map is that the operator (end-user) can *interact* with the system. To make this interaction both possible and efficient, the GIS has to be based on an appropriate data structure. However, most existing systems lack these data structures. We introduce the term *Reactive Data Structure* for a data structure with the following two properties:

- *Spatial organization*: This is necessary for efficient implementation of operations such as: selection of all objects within a rectangle, picking an object from the display, map overlay computations, and so on [10]. Several spatial data structures are described in the literature and are implemented in existing GISs.
- *Detail levels*: Too much details on the display will hamper the operator's perception of the important information. Also, unnecessary details will slow down the drawing process. When the operator wants to take a closer look at a part

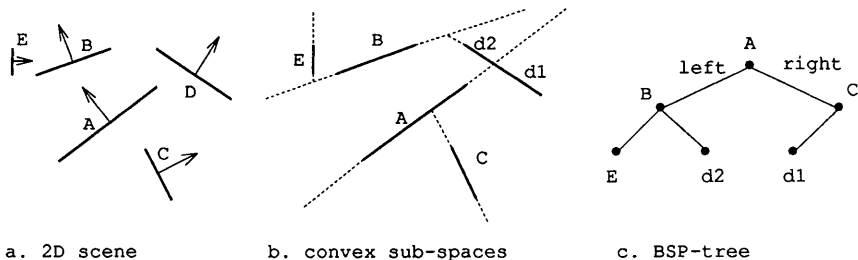


Figure 1: The building of a BSP-tree

of the map, the objects are enlarged, and more details are drawn (new objects). Conversely, when zooming out, fine details are removed from the display. We call this operation *logical zoom* in contrast with the ordinary zoom which only enlarges. There is some literature available on data structures with detail levels, for instance strip trees [1] and multi-scale line-trees [6].

The data structure presented in this paper is a modification of the BSP-tree. A short description of the original BSP-tree is given in section 2, together with some minor modifications for the GIS environment. The next section shows how the basic spatial operations can be implemented efficiently by using a BSP-tree. Section 4 describes the most important difference with the original BSP-tree, the incorporation of detail levels. In section 5, an application of the reactive data structure is given. Finally, the pros and cons are discussed in section 6.

2 The BSP-tree and some variations on it

2.1 The original BSP-tree

The original use of the Binary Space Partitioning (BSP) tree was in 3D Computer Graphics [5, 4]. Figure 1a shows a 2D scene with some directed line segments. A 2D scene is used here, because it is easier to draw than a 3D scene. However, the principle remains the same. The “left” side of the line segment is marked with an arrow. From this scene, line segment A is selected and the 2D space is split into two parts by the supporting line of A, the dashed line in Figure 1b. This process is repeated for each of the two sub-spaces with other line segments. The splitting of space continues until there are no line segments left. Note that sometimes the splitting of a space implies that a line segment (that itself is not yet used for splitting), is split into two parts. D for example, is split into d1 and d2. Figure 1b shows the resulting organization of the space, as a set of (possibly open) *convex* sub-spaces. The corresponding BSP-tree is drawn in Figure 1c. In the 3D case supporting planes of flat polygons are used to split the space instead of lines.

The choice of which line segment is used to split the space, very much influences the building of the tree. We want the BSP-tree to be balanced and have as few nodes as possible. These two wishes are conflicting, because balancing the tree requires that line segments from the middle of the data set are used to split the space. These

line segments will probably split other line segments. Each split of a line segment introduces an extra node in the BSP-tree. It is not clear how we can optimize the BSP-tree, so further research is needed here.

The appendix contains the Pascal code of a program that builds a BSP-tree. The program *BuildTree* is a variation of the traditional method for building a BSP-tree [5]. The procedure *SplitLine* and the functions *LineSituation*, *CreateNode* and *GetLine* are not included in the appendix, because their meaning will be clear. A node in the BSP-tree is represented by the record type *node*, which contains a line segment and pointers to the left and right child. Initially, the tree is empty. As long as *GetLine* can fetch a new line segment, it is added to the BSP-tree with a call to the function *AddLine*. *AddLine* checks whether the correct position in the BSP-tree is found. This is true if the current pointer *tree* in the BSP-tree is nil. In that case a new node is created and added to the tree. Otherwise, *LinePosition* determines in which sub-tree the line segment has to be stored. The storage of the line segment is implemented by a recursive call to *AddLine*. It is possible that the line segment has to be split first.

The splitting of line segments has a serious drawback. If we have m line segments in a scene, then it is possible that we end up with $O(m^2)$ nodes in the tree. It will be clear that this is unacceptable in GIS applications, in which we typically deal with 10,000 or more line segments. However, this is a worst case situation and the actual number of nodes will not be that large.

2.2 The object BSP-tree

The BSP-tree, as discussed so far, is only suited for storing a collection of (unrelated) line segments. In a modeling system it must be possible to represent a closed object; for example (the interior of) a polygon in the 2D case, or a polyhedron in the 3D case. The *object BSP-tree* is an extension to the BSP-tree to cater for object representation. A polygon is defined by a set of line segments, which together make up the boundary of the polygon. The boundary of an object can be stored in a BSP-tree. The BSP-tree is extended with explicit leaf nodes which do not contain a splitting line segment. These leaf nodes only correspond with the convex sub-spaces created by the BSP-tree. A boolean in a leaf node tells whether the convex sub-space is inside or outside the object.

At the University of Leiden we used the object BSP-tree in the 3D graphics modeling system HIRASP [7]. Because of the spatial organization, the hidden surfaces can be "removed" in $O(n)$ time with n the number of polygons in the tree [8]. The object BSP-tree is also well suited to perform the set operations: union, difference and intersection, as used in Constructive Solid Geometry (CSG) systems [9]. The *map overlay operation* in a GIS (described in [10]) can be compared with these set operations.

2.3 The multi-object BSP-tree

We want to exploit the spatial organization properties of the BSP-tree in a Geographic Information System. In a GIS we usually deal with 2D maps. The line segments of the original data base are used to split the space in a recursive manner. By using data

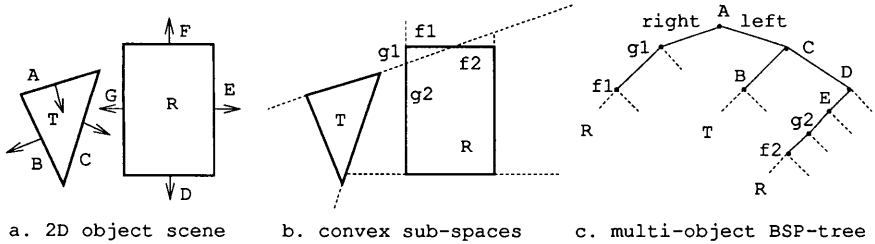


Figure 2: The building of a multi-object BSP-tree

inherent to the problem to organize the space, we expect a good spatial organization. Maps always contain multiple objects; for example several countries on the map of Europe. Because we deal with multiple objects we have to modify the concept of the object BSP-tree. Instead of a boolean, the leaf nodes now contain an identification (name). This identification tells to which object the convex sub-space, represented by the leaf node, belongs. This type of BSP-tree is called *multi-object BSP-tree*.

Figure 2a shows a 2D scene with two objects, triangle T with sides ABC and rectangle R with sides DEFG. The space is organized as a set of convex sub-spaces. The result is shown in Figure 2b. The BSP-tree of Figure 2c is extended with explicit leaf nodes, each representing a convex part of the space. If a convex sub-space corresponds with the “outside” region, then no label is drawn in Figure 2c. A disadvantage of this BSP-tree is that the representation of one object is scattered over several leaves; for example rectangle R in Figure 2. The following list summarizes the properties of the multi-object BSP-tree:

- Each node in the tree corresponds with a convex sub-space.
- Each internal node splits the convex sub-space into two convex parts: left and right. The convex sub-spaces become smaller when the tree is descended. Each internal node contains one line segment.
- Each leaf node corresponds with a convex sub-space which will not be split. A leaf node does not contain a line segment, but it does contain an object identification.

3 The basic spatial operations

In this section we will explain how the (multi-object) BSP-tree is used in implementing two spatial operations: the pick and the rectangle search.

3.1 The pick operation

A map is displayed on the screen. The user selects a point $P = (x, y)$ with an input device such as a mouse or tablet. He wants to know which object he pointed at. To solve this problem we have to locate point P in the tree. This is done by descending

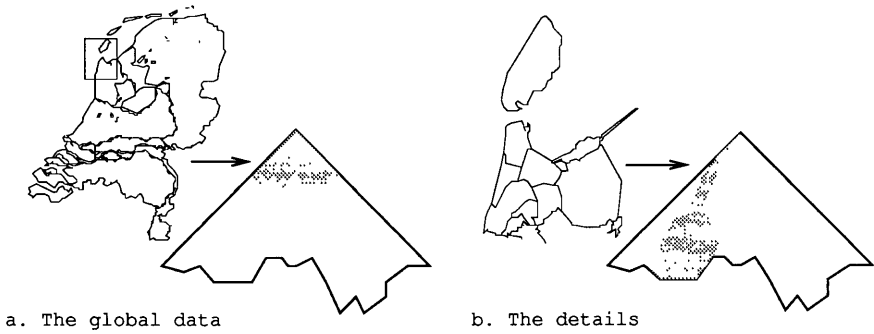


Figure 3: Global and detailed data in the reactive BSP-tree

the tree until a leaf node is reached. This leaf node contains the identification of the object. The descending of the tree is quite simple: if at an internal node point P lies on the left side of the line segment, then the left branch is followed, else the right branch is followed. This results in one straight path from the root to a leaf node. If the tree is balanced and n is the number of internal nodes in the tree, then this search can be performed in $O(\log n)$.

3.2 The rectangle search

The user wants to select all objects within rectangle R . This operation is also performed when (a part of) a map has to be displayed on a rectangular screen. Basically, the traversal of the tree is the same as in the pick operation. At an internal node, the left branch is followed if there is an overlap between rectangle R and the left sub-space. And, of course, the right branch is followed if there is an overlap between the right sub-space and the rectangle R . If there is overlap with both sub-spaces then both branches must be followed. This traversal can be accomplished by a simple recursive function. Only the nodes encountered during this traversal contain objects that are within the rectangle.

The efficiency of these operations is based on the fact that whole parts of the tree are skipped. In an unstructured collection of data we would have to visit every item and test if we “accept” this item based on its geometric properties. Using the BSP-tree we don’t have to examine the data that are outside our region of interest.

4 The detail levels

We need detail levels, as argued in the introduction, if we want to build an interactive GIS. The detail levels must not introduce redundant data storage and must be combined with the spatial data structure. Not only the geometric data must be organized with detail levels, but the same applies to the related application data. However, we will focus our attention on the geometric data.

We first make an observation of the BSP-tree created with the function `AddLine`. A line segment that is inserted early, will end up in one of the top levels of the BSP-tree. A line segment, inserted later on, must first “travel down” the tree (and if necessary

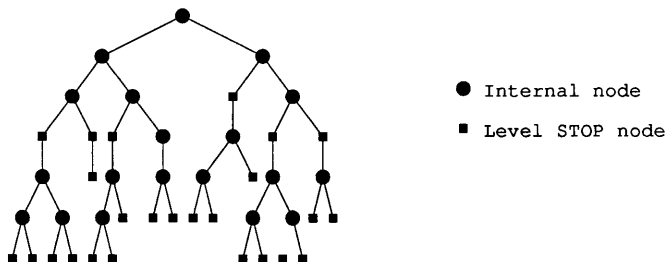


Figure 4: The reactive BSP-tree

be split a few times), before it reaches the correct position and is stored in a new node on a *lower* level of the BSP-tree. We use this property to create a *reactive BSP-tree*. If the global data is inserted first in the BSP-tree, then it will end up in the higher levels of the BSP-tree. The local data (details) are added later, so they end up in the lower levels of the BSP-tree. Figure 3 depicts this situation for a map of The Netherlands. The rectangle in the global map shows the position of the detailed map. The “mountain” represents the whole BSP-tree and the gray region stands for the part of the BSP-tree that contains the data of the corresponding map.

We will use a case to illustrate the way the reactive BSP-tree functions. The case deals with the boundaries of administrative units. In The Netherlands there are six hierarchical levels of administrative units, ranging from the municipalities (the lowest level) to the whole country (the highest level). We store the boundaries of these administrative units in the BSP-tree, starting with the highest level, then the next highest level, and so on. When we display this map, the number of detail levels depends on the scale. That is, if we assume the size of the screen fixed, the size of the region we want to display. The larger the region we want to display, the less detail levels will be shown. A heuristic rule: the total amount of geometric data to be displayed is constant.

The BSP-tree is traversed with an adapted “rectangle search”-algorithm, to display all objects in a certain region up to a certain detail level. The algorithm must know where one detail level stops and where the other begins. This can be achieved by extending the BSP-tree in one of the following manners:

- Add to each node a label with the detail level. If during the traversal of the BSP-tree a detail level is reached that is lower than the one we are interested in, then we can skip this branch, because it contains only data of a lower level.
- After inserting the global data (highest level) into BSP-tree, add special nodes, called *level STOP nodes*, to the BSP-tree. The level STOP nodes contain no splitting line segment and can be compared with the leaf nodes of the multi-object BSP-tree (see section 2.3). Then the next highest level is added to the BSP-tree, again followed by level STOP nodes. This process is repeated for each detail level. Figure 4 shows a reactive BSP-tree with two detail levels.

A drawback of the reactive BSP-tree is that it only supports a part of the map generalization process. Unimportant lines are removed and small regions are grouped,

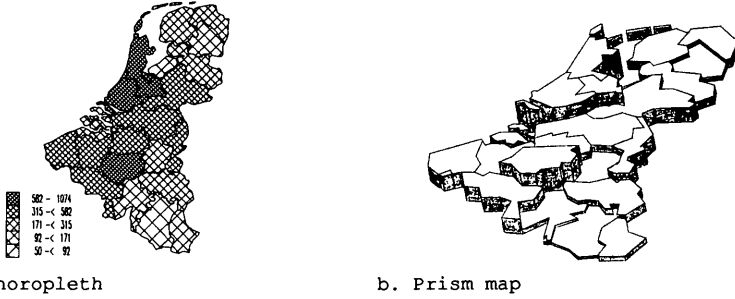


Figure 5: Two map types for thematic mapping

but important lines will be drawn with the same number of points on every scale. As far as we know there is no elegant solution to this problem. It is possible to store a generalized version of a line at every detail level. However, the storage of the same line at multiple levels introduces undesired redundancy. The generalized version of a line can be computed specially for every level with a line generalization algorithm, for instance with the Douglas-Peucker algorithm [2].

5 An application

In this section we describe some additional uses of the reactive data structure in thematic mapping. We will expand the case of the previous section, to make it possible to visualize census data of the administrative units. We show how a choropleth map and a prism map can be produced. An example of those map types is given in Figure 5. The reactive BSP-tree with the level STOP nodes is used. A level STOP node corresponds with a convex part of an administrative unit at that level. The identification of the administrative unit is stored in the level STOP node. The census data is not stored in the BSP-tree, because the BSP-tree scatters objects (administrative units) over several leaves, see section 2. The census data is available at each detail level.

5.1 Choropleth

After the user has decided which region and which census variable has to be displayed, the detail level is determined. On a choropleth map the administrative units are colored. The color depends on the value of the census variable in the administrative unit. All we have to do to produce a choropleth map, is traverse the BSP-tree for the selected region and level. When we reach a level STOP node of the desired level, we know to which administrative unit the corresponding convex sub-space belongs. The required census variable is retrieved and the convex sub-space is filled with the right color.

We do not have an explicit representation of the convex sub-spaces. This is solved by maintaining a *temporary data structure* during the traversal of the BSP-tree. This temporary data structure represents the (open) convex sub-space that corresponds with the current node. Each time we take a step in the BSP-tree the temporary data

structure is updated. We will show that this takes $O(\log n)$. If the BSP-tree is balanced, the height of the tree is $O(\log n)$ with n the number of line segments (nodes) in the BSP-tree. So, a convex sub-space has no more than $O(\log n)$ edges. The insertion of an edge in such a convex sub-space (polygon) takes $O(\log n)$. Displaying the whole BSP-tree while coloring the convex sub-spaces takes $O(n \log n)$ time, because there are n nodes in the BSP-tree. It is possible to store the explicit representation of the convex sub-space in the level STOP node. This reduces the time to generate the choropleth to $O(n)$, but increases the storage requirements.

5.2 Prism map

The prism map [3] is an attractive map to look at and it offers the possibility to display an extra variable by the height of the prisms. Before the prism map is generated the user has to indicate from which direction he wants to look at the prisms. A prism map can be compared with a set of 3D-objects.

Basically the prism map is produced in the same manner as the choropleth map. Instead of coloring the convex sub-space, we lift it up to the desired height. If the convex sub-space has k sides, then each side will result in a 3D rectangular polygon. Together with the top of the prism, this results in $k + 1$ 3D polygons, which must be displayed. Before a polygon is displayed it is projected from 3D to 2D, in order to calculate the actual coordinates on the screen. A number of convex prisms form one prism on the map, just as the same convex sub-spaces form together the administrative unit. This means that the “internal” sides of the prism need not be drawn. We can recognize these if we label those sides of the convex sub-spaces, that are part of line segments. This can be done without overhead.

The “hidden surface” problem is usually quite difficult and time consuming to solve. However, if we slightly change the manner in which the BSP-tree is traversed, the hidden surface problem is solved (in combination with the Painters-algorithm). The different traversal does not cost any extra processing time and ensures that prisms further away from the viewing point are drawn first. This results the “removal” of the hidden surfaces in the prism map, for more details on this topic see [8]. Normally, the whole BSP-tree is traversed in $O(n)$, but we have to maintain the temporary data structure that contains the explicit representation of the current convex sub-space. So, we can produce a prism map in $O(n \log n)$. If explicit representations of the convex sub-spaces are stored, then a prism map can be produced in $O(n)$, which is quite fast. Because of this fast response, the end-user will be stimulated to take other views of the data.

6 Conclusion

The data structure we presented is one of the few that combines the two difficult requirements: spatial organization and detail levels. The reactive BSP-tree fulfills, up to a certain extend, those requirements. The BSP-tree also introduces some problems, as we have seen in the previous sections. And we can think of more problems: How should a single unconnected point be stored in the BSP-tree? How can the BSP-tree be balanced? How will the BSP-tree behave if we insert very large amounts of

irregular geometric data? The BSP-tree is a static data structure. This is not really a problem, because the maps in most GIS applications are also static. In order to gain experience we are currently working on a prototype GIS that is based on a BSP-tree. We are interested in the size and the performance of the BSP-tree. We know that the BSP-tree is far from perfect, but we hope that it serves as a source of inspiration to generate more ideas. A reactive data structure need not be based on a BSP-tree, other solutions are possible. We are also working on development of a reactive data structure based on an Object-Oriented approach to GIS.

References

- [1] Dana H. Ballard. Strip trees: A hierarchical representation for curves. *Communications of the ACM*, 24(5):310–321, May 1981.
- [2] D.H. Douglas and T.K. Peucker. Algorithms for the reduction of points required to represent a digitized line or its caricature. *Canadian Cartographer*, 10:112–122, 1973.
- [3] Wm. Randolph Franklin and Harry L. Lewis. 3D graphic display of discrete spatial data by PRISM maps. *ACM Computer Graphics*, 12(3):70–75, August 1978.
- [4] Henry Fuchs, Gregory D. Abram, and Eric D. Grant. Near real-time shaded display of rigid objects. *ACM Computer Graphics*, 17(3):65–72, July 1983.
- [5] Henry Fuchs, Zvi M. Kedem, and Bruce F. Naylor. On visible surface generation by a priori tree structures. *ACM Computer Graphics*, 14(3):124–133, July 1980.
- [6] Christopher B. Jones and Ian M. Abraham. Line generalisation in a global cartographic database. *Cartographica*, 24(3):32–45, 1987.
- [7] Wim J.M. Teunissen and Jan van den Bos. HIRASP a hierarchical interactive rastergraphics system based on pattern graphs and pattern expressions. In *Eurographics*, pages 393–404, 1988. Amsterdam, The Netherlands.
- [8] Wim J.M. Teunissen and Peter J.M. van Oosterom. The creation and display of arbitrary polyhedra in HIRASP. Technical report, University of Leiden, July 1988. Department of Computer Science, Report 88-20.
- [9] William C. Thibault and Bruce F. Naylor. Set operations on polyhedra using binary space partitioning trees. *Computer Graphics*, 21(4):153–162, July 1987.
- [10] Peter van Oosterom. Spatial data structures in Geographic Information Systems. In *NCGA's Mapping and Geographic Information Systems*, 1988. Orlando, Florida (In press).

A Appendix: BuildTree

```
program BuildTree;

type LineSegment = record x1, y1, x2, y2: real end;
   Pos = (LEFT, RIGHT, SPLIT);
   BSP = ^node;
   node = record
       segm: LineSegment;
       l, r: BSP {Left end Right child in BSP-tree}
   end;

var root: BSP;
    newsegm: LineSegment;

procedure SplitLine
(tree: BSP; segm: LineSegment; var Lsegm, Rsegm: LineSegment); forward;
function CreateNode(tree: BSP; segm: LineSegment): BSP; forward;
function LinePosition(tree: BSP; segm: LineSegment): Pos; forward;
function GetLine(var newsegm: LineSegment): boolean; forward;

function AddLine(tree: BSP; segm: LineSegment): BSP;
var Lsegm, Rsegm: LineSegment;
begin
    if tree = nil then {Position found in BSP-tree, create node}
        tree := CreateNode(tree, segm)
    else begin {Position not yet found, go further down the tree}
        case LinePosition(tree, segm) of
            LEFT: tree^.l := AddLine(tree^.l, segm);
            RIGHT: tree^.r := AddLine(tree^.r, segm);
            SPLIT: begin
                SplitLine(tree, segm, Lsegm, Rsegm);
                tree^.l := AddLine(tree^.l, Lsegm);
                tree^.r := AddLine(tree^.r, Rsegm);
            end
        end
    end;
    AddLine := tree;
end;

.
.
begin
    root := nil;
    while GetLine(newsegm) do root := AddLine(root, newsegm)
end.
```

CHALLENGES AHEAD FOR THE MAPPING PROFESSION

J.C. Muller
International Institute for Aerospace Survey
and Earth Sciences
350 Boulevard 1945, P.O. Box 6, 7500 AA Enschede,
The Netherlands

ABSTRACT

Cartography is in a stage of revolutions. The multi-purpose maps of yesterday, essentially descriptive, static and deterministic are now conceptually challenged by new map products which are extremely volatile, single purpose and probabilistic. The realm of cartographic activities has expanded towards the portrayal of highly abstract geographical spaces. The traditional function of maps as spatial storage device is on the decline, whereas their communication function and analytical power are increasingly emphasized.

Parallely, the multiplication of 'do-it-yourself' micro-computer mapping kits, the emergence of cartographic expert systems and the incorporation of mapping activities within the wider realm of technologies for geo-information production are challenging the integrity of the cartographic discipline.

An attempt is made to investigate the origin of these recent developments and to speculate on the practical and theoretical implications for the mapping profession.

INTRODUCTION

Maps are tools to acquire, analyze and communicate spatial knowledge. Their history is closely related to the history of man who always felt the need to increase his knowledge of the surrounding spaces in order to better exploit the resources required for his survival. For a major part of this history, maps have been used as spatial records of planimetric and, more recently, topographic information. Maps were mostly viewed as symbolic representations of the visible space, with various degrees of abstraction. The map products were essentially descriptive, static and deterministic. Map queries were geometrical and related to place locations, distances, orientations, areas and elevations.

With the advent of computer, remote sensing and geographic information systems, the nature of maps has undergone a dramatic change, however. Most of the maps of today are thematic maps, which emphasize the attributes of places rather than their location. They often give a probabilistic view of physico-socio-economical phenomena that are not thoroughly commensurable. They portray a temporary view of a world which is changing faster than it takes to produce them. Finally, their look is more than ever influenced by the use of highly versatile

sophisticated graphics which may enhance or destroy their power of communication.

It would be erroneous to think that these new developments were strictly a result of the computer revolution. Computer technology only provided the means for such transformations to occur. In the history of science, problems often appear to come along with their solution. In this case, computers were used to bring about a new cartography capable of providing solutions to the problems of managing large amounts of thematic and topographic data. Following are examples of the new challenges faced by the cartographic profession.

INFORMATION EXPLOSION

We live in an information society. In the United States it has been estimated that more than fifty percent of the Gross National Product stems from information-related areas and less than fifty percent from manufacturing. Furthermore, it has been estimated by different investigations, that annually, somewhere between 9×10^{11} and 2×10^{12} images are printed world-wide. This means about one picture per day per capita. Hence, the trend towards non-verbal communication is strong and is increasing. Part of those images are used to communicate scientific information and appear in the form of statistical graphics and maps. Specifically, there is a need to portray an increased amount of spatially referenced data which has expanded beyond the narrow limits of the portrayal of the surface of the Earth. We are at a time of new explorers who do not travel across the oceans and continents, but who explore and collect data about the fabric of our physical and socio-economical environment. They are busy stacking up information related to the subsurface of the earth, the atmosphere, the geography of populations and socio-economical activities. The value of this information increases with the worldwide increased human pressure on the environment, the disruption of the earth ecosystem, the urban growth and the threats of destruction of our resource base. This information, however, needs to be organized, analyzed, and interpreted in order to be used by the planner, the resource manager, or the politician. An essential part of this "digestion" process is done through visual analysis of spatial data depicted in the form of maps. In this respect, the cartographic profession faces two challenges:

- 1) how to respond economically and efficiently to those map needs at a rate which usefully follows the extraordinary growth of digital data bases, and
- 2) how to provide an optimal answer to the problems of rapid updating at a time where the life cycle of environmental and socio-economical data is often shorter than the map production cycle.

DEMOCRATIZATION OF THE DECISION MAKING PROCESS

The principle of delegation of power and decision in democratic societies can only be implemented if there is a liberal access to information. In turn, this requires a distributed network of information sources and communication which are available at the various levels of government. Modern technology provides the means for data communication through satellites, telephone lines, cabling and fiber optics. Furthermore, micro-computer based geographic information and mapping systems now give the possibility to analyze the data and produce the maps at the lowest level of authority where the interaction between the mapper and the user can take place most efficiently.

A challenge for the surveying and cartographic profession is to provide the guidelines which are required for efficient data exchange and communication. Another challenge is the delegation of responsibility for data maintenance. Traditionally, national mapping agencies have been responsible for the creation and maintenance of national mapping programmes. Provincial and municipal authorities are now custodian of their own maps as well. Should they be left alone in insuring the updating and maintenance of their own data? Do they have the financial means to do so? One must find a formula which insures a firm control at the national level for data integrity and maintenance with the necessary degelation of power and accountability at the local level as well.

Another aspect of the democratization process is the increased liability of local governments for the management of their own resources. The saying goes "local problems are best solved locally." The recent spread of geographical information systems at the municipality level illustrates this trend. The distinct function of maps in the GIS tool box is of particular significance for cartography. For the first time in the long history of mapping is the map systematically recognized as a procedural tool for the simulation, modelling and interpretation of spatial processes. Some investigators have gone so far as suggesting the emergence of a new map era: the algebra of maps or parametric mapping. Maps are variables represented as organized sets of numbers and, like in traditional algebra, can be added, subtracted, exponentiated and logically sequenced to form equations. The spatial coincidence and juxtapositioning of values among and within maps create new operators, such as proximity, spatial coincidence, and optimal paths (Berry, 1987). GIS technology has revolutionized the way we handle or use maps. Along with other professionals, cartographers have shifted their dependence from pure graphics to an increasing dependence upon digital databases and spatial theories. Part of the cartographic community may have some difficulties in accepting those changes, particularly for those cartographers who view themselves essentially as symbolic designers or graphic

communicators. But this revolution merely goes back to the traditional geometrical roots of cartography with the added power of matrix algebra and spatial statistics.

The use of maps as spatial operators in the decision making process is not without problems, however. Many spatial relationships cannot be adequately quantified. A theory for spatial statistics is only emerging, and one does not know adequately the effect of spatial uncertainty in the manipulation and the combination of maps.

INCREASED ABSTRACTION OF SOCIAL/ECONOMICAL COMMUNICATIONS AND MOVEMENTS

Cartographers are used to deal with non-Euclidean geometries to resolve the equation of the spherical surface onto a plane. The mapping of socio-economical spaces is much more elusive than the mapping of the earth surface, however, and goes outside of the Gaussian or Riemannian classical differential geometries. The notion of distance between people and places is very much depending on the means of transportation or communication and is rarely consistent with the measure of geographical distance. Time distances, cost distances or psycholocial distances are often prevailing over geographical distances and require to be mapped in order to understand and predict people's behaviour. There has been much study but little success in the cartography of socio-economical distances, for the obvious reason that the traditional map format which implies a continuous metric space is not well fitted to the metric inconsistencies and discontinuities in the numbers of hours, dollars or kilowatts that separate people and determine people's interactions.

Cartographers face here a new challenge which is more than just an intellectual curiosity: our ability to understand, control and monitor the multilayers of space which make up the fabric of human relationships will depend on the means of analysis and interpretation which are made available to the planner and the decision maker. New forms of data portrayal will have to be invented for this purpose. Geometrical, topological and thematic relationships between spatial elements will have to be encoded and structured in such a way as to allow for easy and fast spatial queries, spatial analysis and conceptual generalization based on context rather than single geographical objects.

It is interesting to note that the advent of digital cartography, for the most part, has done little to promote new cartographic products. Most efforts in the development of cartographic software have been directed toward the emulation of manual cartographic products of high graphic standards (Goodchild, 1988). That is, we still think of a spatial representation as something pressed flat on the finite dimension of a piece of paper

or the screen of a graphic station. We still think of a map as a visual product where graphic appearance holds priority over content in data quality and resolution. A challenge for the cartographer will be to extend his activities towards the manipulation of the invisible part of the landscape model, where data are unclassified and ungeneralized, where lines are fuzzy and areas heterogenous, within the framework of a multidimensional space including time. A stronger involvement of cartographers in "pre-cartography" will emancipate cartographic thinking as dictated by a platonic view of a world made of ideal forms and help design cartographic products where "what you get you cannot see".

Recent developments that will contribute to that emancipation are the introduction of raster technology (in which the physical domain is reduced to atomic objects), hierarchical data structures leading towards multiresolution representation and the advent of dynamic three-dimensional stereo perspectives on a Tektronix screen which allows the eye to look "behind" the scene.

WIDE ACCESS OF TURNKEY CARTOGRAPHY WITHOUT CONCOMITANT SPREAD OF CARTOGRAPHIC KNOWLEDGE

A new phenomenon has emerged recently which may be potentially disruptive to the all evolution of cartographic science: the multiplication of commercial micro-based computer packages which allow the user to make his own images without the costly help of a cartographic expert. The reduction in cost of computer hardware for the production of graphics and the recent development of mini-based computer cartography software are the main reasons for this upsurge. On the one hand one may welcome such development, as it will enhance the role of maps in the analysis and communication of spatial information by extending the use of cartographic products to a wider circle of society which normally could not afford them. Such development also reinforces the present trend toward customized cartography where maps are produced for a few individuals in order to respond to specific queries, at a time when the social role of general purpose topographic map series appears to be on the decline. Customized maps are typically very specialized and have very short life cycles. They are more like working maps which are thrown away once the problem at hand has been resolved. Hence, they must not be very expensive, they must be quickly designed and strictly limited in scope.

The emergence of micro-based computer packages apparently provides a technological response to those requirements. The user can now produce his own specialized map quickly and cheaply. Hence the traditional separation between map makers and map users is disappearing. One may expect that the quality of maps and the evolution of cartography as a tool for spatial analysis and communication will be greatly affected by this new generation of self-made map

makers who have had no cartographic training but who have the tools to make maps through simple turnkey devices. As a result, one could think that cartographers will loose control of the evolution of their own discipline. Several indications point towards this direction. Cartographers have had so far little input or control in the production of micro-computer cartographic software, whose main developers are coming from the computing science disciplines. The user of those packages usually has no interest in cartographic principles and is only interested in learning the operating system which will allow him to produce the map which we believe will best fulfil his needs or interests. There are practical and philosophical implications related to this new development.

The basic models of cartographic communication, developed in the seventies when the production of maps was still monopolized by cartographers, will have to be changed. The user is not only the receiver, but the transmitter as well. The social and cultural biases of the user's mind will occur upstream in the map communication process. One might further argue that noise will be reduced, since one (the cartographer) does not talk to somebody else (the map reader) but rather talk to himself.

The danger, of course, is the loss of the so-called "neutrality" of the expert cartographer who was striving for scientific integrity. An additional problem is the lack of cartographical skills. I am not referring here to the ability to draw lines and symbols, but to the various domains of knowledge which are required for designing a map, including a basic understanding of geography, spatial processes and graphicacy. Our secondary school curriculums usually give little training in those areas. Hence, the user will be provided with a tool which he probably does not know how to use. Since one cannot undertake the cartographic training of the thousands of potential users who are going to improvise map making, we will have to research the possibilities of implementing 'intelligent' front ends to the cartographic execution programs, preferably in collaboration with the commercial vendors. This front end may take the form of a cartographic expert system which emulates the expertise of a human cartographer and provides guidance to the naive user. Knowledge engineering of cartographic expertise in order to build the knowledge base of such systems will not be easy. It forces cartographers to analyze and to formalize the decision process which leads to the making of a map. Hence academic cartography will be forced out of the textbook to be confronted against judgemental rules which must be agreed upon by everyone. There is some doubt as to whether or not a consensus can be reached about the procedural aspects of map design, which is essentially a subjective, nonrepetitive and hollistic process. It seems that only the simplest aspects of graphic semiology can be agreed upon, such as the adequation between measurement scale and graphic

variable, for which we do not really need an expert system! Perhaps developing "negative" ("idiot proofing") systems to discourage dangerous practices is a more realistic undertaking than prescribing mapping choices, since it seems easier to agree on what should not be done. The idea of idiot proofing systems for GIS applications has already been suggested by David Rhind (1988).

CONCLUSION

One can observe several trends in the recent development of cartography, both in response of the global demands for spatial information as well in promoting new needs for information by increasing the public awareness that spatial information may be obtained and can be used for planning and policy making purposes:

- 1) At a time when the social and economical relevance of the traditional large scale, multipurpose analogue mapping series is increasingly hard to justify, thematic map products, in analogue or digital forms, are rapidly expanding. The simultaneous growth of remote sensing and GIS technologies and the extraordinary development of national and regional census agencies are partly responsible for this expansion.
- 2) The life cycle of maps is becoming increasingly shorter. Up-to-date maps are required to portray the rapid changes in economic development, resources, environmental pollution and urbanization.
- 3) The traditional function of maps as spatial storage device is on the decline, whereas their analytical and communicative functions are now stimulated in another form through the introduction of GIS technology and high quality computer graphics packages.
- 4) New forms of spatial representations are being investigated to depict non-metric spaces, subsurfaces, the time dimension and the fuzziness of spatial objects.
- 5) GIS technology has revolutionized the way we use maps. Maps are now systematically used as spatial operators to analyze and experiment the interaction between the various layers of physical and human geography.
- 6) Maps are increasingly produced for one purpose for one single user. This trend towards customized mapping was made possible through the combination of computer-assisted cartography, computer graphics, remote sensing and GIS technologies.
- 7) The traditional separation between map makers and map users is disappearing, with the map client being able to produce his own maps on demand with a computer cartographic package. Hence, the models of cartographic communication developed in the last twenty years are now somewhat irrelevant. The major issue is whether the user can communicate to himself via the map channel which he creates.

There has been too little time to appreciate the effect of these recent trends on the theoretical fabric of the cartographic discipline. Where will cartography be ten years from now? With the GIS and remote sensing revolutions, maps appear to be losing pre-eminence. They are increasingly viewed as by-products of technologies for geographic information extraction, modelling and interpretation. Some even doubt "whether an entire discipline will still exist to make and study [maps]...; most likely expertise for creating maps will reside within the tool-creating enterprise that is increasingly being referred to as "spatial data handling" (Petchenik, 1988). Others fear that "automated mapping techniques which could be used as a tool to multiply human cartographic capability, may evolve increasingly into tools which are independent of humans" (Gallant, 1987). This statement refers to the (unrealistic) prospect of full-fledged cartographic expert systems. The present trend towards "do-it-yourself" customized maps on demand may appear threatening to the professional cartographer, but so did photogrammetry for the land surveyor. As professions are being challenged, they are forced to critically review their activities, expand their skills and explore new areas of endeavour. The remarkable development in recent years is the increasing awareness of a communality of goals between surveyors, photogrameters, cartographers, geographers and other spatial scientists towards the production of geo-information. The joining of forces appears to be an obvious answer to the challenge of survival, as each discipline is individually incapable to cover the scope of technology and knowledge which is now required for the production of geo-information.

The research agenda for automated cartography in the 1990s, therefore, will have to include many disciplines. Fundamental questions, such as the following, will have to be answered: (1) What kind of information can be imparted to cartographic expert systems? (2) What are the criteria and priorities for developing cartographic software in a desk top publishing environment? (3) How can we handle the time dimension in our cartographic displays? And (4) How can we operationalize the process of automated map generalization in a holistic fashion? A comprehensive research model that will integrate the many technologies involved in the production of geo-information is required. Such a model would act as a clearing house for knowledge, and it would eliminate duplication of ad hoc research initiatives undertaken on either side of discipline boundaries. The identity of the cartographic discipline may become increasingly fuzzy in the process of this evolution, but the role of maps as modelling tools will become potentially more valuable.

REFERENCES

- Berry, K. Joseph. 1987, Computer-Assisted Map Analysis: Potential and Pitfalls: Photogrammetric, Engineering and Remote Sensing, Vol 53, pp 1405-1410.
- Gallant, Q. Davic. 1987, The Misguided Evolution of Future Mapping Technology: Proceedings Autocarto 8, pp 386-395.
- Goodchild, F. Michael. 1988, Stepping Over the Line: Technological Constraints and the New Cartography: The American Cartographer, Vol. 15, pp 311-319.
- Petchenik, B. Barbara. 1988, Afterword: The American Cartographer, Vol. 15, pp 321-322.
- Rhind, David. 1988, A GIS Research Agenda: International Journal Geographical Information Systems, Vo. 2, pp 23-28.

AN ON-LINE, SECURE AND INFINITELY FLEXIBLE DATA BASE SYSTEM
FOR THE NATIONAL POPULATION CENSUS

D.W. Rhind, E. Hayes - Hall, H.M. Mounsey¹ and S. Openshaw²

1 : Department of Geography, Birkbeck College, University of
London, 7-15 Gresse Street, London W1P 1PA, UK

2 : Department of Geography, The University, Newcastle, UK

ABSTRACT

This paper describes a prototype on - line data base system to handle the 20 million or more records which will arise from responses to the individual questionnaires for the next Census of Population in Britain in 1991. The work is funded by the UK census agencies and by the Economic and Social Research Council. Unlike conventional census output, it is predicated upon producing the necessary results 'on demand' for variables and areas specified by users in a variety of ways over national telecomms networks. The methods of achieving this have involved use of two quite different computer systems, one being primarily a hardware - based solution and the other a more general, software - based approach. These two approaches are described, together with the means of maintaining confidentiality in the 'raw' data (as required by statute) and the implications of operation of such a system for value - added census services.

INTRODUCTION

Statistics from the decennial Census of Population form possibly the single most widely used data source in Britain (DoE 1987). Traditionally, these statistics have - like those from most other censuses - been produced as area aggregate statistics in count or cross - tabulated form (Redfern 1987, Dewdney and Rhind 1986). These have described the overall characteristics of the people or the households in each areal unit and their form has had to be planned long in advance of the census itself (see various chapters in Rhind 1983). In Britain, the Small Area Statistics (SAS) are the best example of such a pre-defined set of tables; inevitably, these reflect compromises after discussions with many potential users of the statistics (Denham and Rhind 1983). In contrast, unanticipated requirements are met by special tabulations, charged and produced to the specification of individual customers, largely after the production of the standard census statistics. Distribution of all these statistics has hitherto been in paper form or on magnetic media.

Two census - taking agencies exist for mainland Britain : these are the Office of Population Censuses and Surveys (OPCS) in England and Wales and the General Registrar's Office for Scotland (GRO(S)). Collectively, they have long recognised that this dual form of output is scarcely ideal. Aided by increasing densities and diminishing costs of computer storage, they have sought to provide a wider range of information in the standard products. The 1971 Census SAS for England, for instance, consisted of only 1,571 cells of cross-tabulated information derived from 30 questions whereas

the 1981 equivalent consisted of 4,400 cells from only 21 questions. It has recently been proposed that the standard tabulations for the 1991 Census should contain about 6,000 cells (OPCS 1988). This 'shot gun' approach has had a price; as analyses of Census data used by active user organisations such as Tyne and Wear County Council have indicated, any one organisation may never use many of the cells and some users have been unable to afford or wait for the special tabulations. Finally, the volume of cross-tabulated summary statistics has come increasingly close to those of the 'raw' data : the most detailed 1991 SAS data, suitably compressed, are likely to occupy about 750Mb whilst bit-compressed 'raw' data could amount to between 600Mb and 1Gb.

This approach to census data processing stemmed from 1960's and 1970's technologies; it led directly to the advent of a single package commissioned by a consortium of (mostly local government) users to handle the 1981 SAS; the result, called SASPAC, ran on many different ranges of computers and operating systems and was arguably the most portable product of its day. It was responsible for introducing perhaps 2,000 individuals to census analysis (Rhind 1984). Despite SASPAC's success, the software was a creature of its time, being necessarily tailored to virtually the lowest common denominator of computing availability. The institutional situation at that time was typified by the sole availability of stand-alone mainframe computers run by centralised administrations, by batch access, by relatively restricted and 'unfriendly' software and by a punched card record orientation.

This world has changed totally. Desk top micros selling for around \$1,000 now have far more data storage and processing capabilities than many of the mainframes in use in many local authorities in 1981. In particular, the entire SAS data for all 130,000 Enumeration Districts (EDs - the approximate equivalent of Collector's Districts) in Britain could, for instance, now be stored on a CD-ROM of the type which the UK Post Office are now selling: their price of \$7,000 for hardware reader and disc containing details of 1.5 million unit postcodes indicates the fundamental nature of the change in the last six years - particularly since the price level set reflects the UK government's desire to maximise its financial return on data collated under its auspices. In November 1988, the census offices revealed plans by a number of commercial firms to distribute 1981 SAS on such media and many other national (e.g. the Swedish Land Survey) and international agencies (e.g. World Data Centre A in Boulder, CO.) and commercial agencies are already distributing data by these methods.

Moreover - in universities and some businesses at least - the use of computer networking to move files or to access specialist software on remote computers is now an everyday activity. Distributed databases and distributed computing power is becoming a reality and recognised as such - the Computer Board for the Universities in the UK (which is responsible for approving all central planning of computing facilities in all British universities) argued this to be the most far-reaching change in computing in its three yearly report published in December 1988. As another example, the

feasibility of running a three site US National Centre for Geographic Information and Analysis and an eight member, 13 site set of Regional Research Laboratories in the UK is substantially dependent upon routine use of electronic mail and remote access to software and data bases.

There is every reason to believe that these technical and technological developments will continue and will spread to other sectors. Computing facilities in 1991, then, will be unrecognisable as compared to those of a decade earlier.

SOME NEW PROPOSALS FOR CENSUS DATA HANDLING

Given all this, Rhind (1985) argued for an equally fundamental review of census strategy. He outlined two options, the more radical of which saw the census mainly as a calibration tool with a wider range of data being assembled from various other sources and linked together. The second option was one in which OPCS/GRO(S) would produce only the most basic summary statistics and distribute these, say, on floppy disc for PCs or their 1991/2 equivalent; it is certain that much competitive software will be available to read, analyse statistically, tabulate and map a set of standard statistics for each areal unit. In tandem with this, an on-line service was proposed to be provided by OPCS/GRO(S) for querying a data base of unaggregated Census records, from which users could extract any desired aggregate information - for whatever area(s) or groups of people and cross-tabulated as required - unless this was liable to disclose details of an identifiable individual or household or lead to unreasonable intrusions into privacy. It was further proposed that this would be achieved through building up an Intelligent Knowledge Based System incorporating whatever rules OPCS/GRO(S) already applied and whatever others were found to be necessary.

Following discussions, OPCS and GRO(S) provided a contribution of £11,000 towards the funding of a project to investigate the feasibility of such a system and preliminary work began in late 1986. The funding by OPCS/GRO(S) of course, does not imply any commitment to introduce such an on-line system, should it prove feasible. Following a further grant of £22,000 from the ESRC from November 1987, work accelerated. By April 1988, progress was sufficient for an initial demonstration of the system and for a clause referring to possible use of such a tool to be included in the White Paper on the Population Census (HMSO 1988 para 53).

PROJECT OBJECTIVES

The objectives of the research project were:

- (i) to create a prototype on-line storage and retrieval system capable of handling Census questionnaire returns from the 1991 Census. This would permit :
 - users to specify which counts or cross - tabulations they require at any particular level of aggregation of variables and / or geographical area. So far as

geography is concerned, we have assumed that at least three ways of defining areas of interest are required : using the standard census nested area hierarchy (ED/Ward/District/County/Region/Country) and any variant of it (e.g. ED/Health District/Health Region) as 'building blocks'; equivalent hierarchical and ad-hoc combinations of unit postcodes; and use of National Grid References. This pre-supposes multiple indexing. So far as census variables are concerned, we assume all possible combinations of responses to census questions within one return should be feasible.

- users to obtain an estimate of cost prior to the execution of the job,
 - confidentiality of the 'raw' data to be maintained,
 - print out of all standard SAS and other tabulations as well as the 'special ones'.
- (ii) to create 'pseudo 1981 Census' questionnaire responses for sufficient areas to test the system adequately.
- (iii) to demonstrate the system described in (i) on data derived from (ii).
- (iv) to document the results, including a summary of the aggregation and data release rules finally agreed with OPCS and GRO(S), the level of 'noise' required to maintain confidentiality, the results of user trials and predicted implications of introducing such a system into operational use.

For reasons which are described below, the initial approach was developed only to the stage where it demonstrated that many of the project objectives could be met. The approach was based upon the creation of a superstructure to the ICL INDEPOL system : this exploits the high security and speed of access provided by ICL hardware, the Content Addressable File Store or CAFS (Carmichael 1986, Wise and Pellett 1988). ICL facilities have been in use by OPCS and GRO(S) for 30 years and CAFS is a standard facility in all ICL 3900 mainframes. However, following a review of OPCS' IT strategy by a firm of management consultants, it was decided to base future strategies on an initial choice of a software environment, rather than selecting the hardware and then obtaining or creating the necessary software. Model 204, a relational database with such features as the Structured Query Language (SQL), was selected to form the main part of the new OPCS software environment. Since there is no current implementation of Model 204 on ICL equipment, this decision by the census offices - taken for much broader considerations than any one particular tool (especially one being developed in a research study) - necessitated an equipment procurement which might well preclude use of our ICL - specific system.

Faced with this, we opted to mimic the facilities already developed on the ICL equipment by re-implementing them (so far as possible) under Model 204. Fortunately, we found one

university installation in the country using the software and were kindly allowed access to the software running on the University of Manchester Regional Computer Centre. At the time of writing, OPCS had not finalised which hardware and software would be used for the entirety of the 1991 Census data processing; a mixed ICL/Model 204 configuration seemed a real possibility. For this reason, this paper contains sections on the technical aspects and a preliminary assessment of the relative merits of each of the two approaches.

THE TEST DATA

The data base required to test and to demonstrate the resulting prototype software was the initial priority in the project. 'Raw' data from recent censuses are not, of course, available because of the statutory constraints embedded in the 1920 Census Act; this precludes the release of data pertaining to identifiable individuals in less than 100 years after its collection. The obvious solution of coding up 1881 or earlier records was rejected because of the resource implications. The data set finally used is genuine, 'real' data and contains variables which approximate closely to those collected in GB in 1981 : therefore it is a near-ideal solution to the problem. These data were made available to us as a result of previous work in classifying individual - level census data carried out by one of the authors (SO) and colleagues at Newcastle University for a census agency in another European country.

A data base pertaining to 481,165 households and 1,228,068 non-UK individuals has been set up on disc. Details in the population file include unique id., census (ED) area code, age, sex, marital status and occupation of the individual concerned. For the household file, the details stored include unique id., census area code, locality, occupancy, tenure, number of rooms, and presence or absence of toilet, flushing toilet and bathroom. Since the great bulk of accesses to census data are on a within - county basis (indeed the data are distributed by OPCS on a county by county basis), it is logical to partition the data set on this basis and the files created approximate to the characteristics of a 'typical' English county. Thus the test data set provides a good approximation in access times, though not in total data volumes, to a national data base.

AVOIDING DISCLOSURE OF CONFIDENTIAL INFORMATION

Central to any success that our system might have is the ability to guarantee that no information will be disclosed on identifiable individuals. Failure to ensure this would render impossible the use of the system by users outside of OPCS, at least directly over telecommunication lines. In addition to a need for the technical demonstration of security, the public perception of such access to confidential data must also be satisfactory. This paper is, of course, mainly concerned with the technical problems but plans were made for independent testing of the system (see objective (iv)).

The problem of non-disclosure has been treated both

theoretically and empirically (see, for instance, Duncan and Lambert 1986, forthcoming; Bethlehem, Keller and Pannekoek 1988; Redfern 1988). So far as this project is concerned, our objective is to release no aggregate data which are such that a single individual can be identified and characterised, even by repeated querying of the data base and subtraction of the results of slightly overlapping queries. We took a policy decision that the prevention of disclosure could not be based upon comparison of requests with all previous requests (though maintaining a log of all requests is part of our system); thus, all statistics must be produced in such a fashion that security is preserved irrespective of any comparison made with previous output. More positively expressed, we seek to do the minimum to the data which will retain confidentiality, thereby maximising the value of the data. Strictly, the problem of identification of any one individual from a micro-data set (such as the US Public Use Sample) is not our concern unless this system is used for generating (perhaps purpose-specific) microdata sets.

Four obvious techniques exist for ensuring that no disclosure occurs in aggregate data of the kind hitherto released in Britain :

- deletion of data where the value(s) of some variable(s) do not exceed a defined threshold
- banding of the data into groupings (e.g. with counts ending only in 0s or 5s)
- the introduction of 'noise' by adding small random numbers and the publicising of this element of randomness
- adjusting the area of aggregation in response to the frequency distribution(s) of interest.

Of these, the first and third have been used routinely in the UK in recent censuses and, in the manner implemented, have been accepted without significant debate. In the Small Area Statistics, for instance, it has broadly been the practice to provide missing data values for all bar the most basic few cell counts unless the resident population was 25 or more people (population records) in the area concerned or unless there were 8 or more households (in the household records) therein. All cell counts other than those few remaining in suppressed records were then subject to adjustment through the addition of +1, 0 or -1 (Denham and Rhind 1983, p. 81). The amalgamation of adjacent areas to produce statistically sound and confidential aggregates has also been followed by OPCS in dealing with the 10% sample data and with the 1971/81 Change Files.

Precedent is a valuable matter in dealing with confidentiality : moreover, we know of no circumstances where this has been breached in previous censuses. Hence, our initial experiments have used data deletion and addition of 'noise' and have adopted, initially at least, the same thresholds as OPCS' previous practice. The effect of the application of such rules is, of course, highly data-dependent. Of the 125,000 Enumeration Districts in the the

1971 SAS, the data for only 800 suffered suppression whilst of the corresponding SAS data for the 149,000 populated lkm grid squares, no less than 54% contained suppressed fields (though these related to less than 5% of the total population). This difference reflects the difference between the low variance in population in areal units chosen to have approximately equal numbers to even out the enumerator's workload and the massively skewed distribution in geometrically regular areas. Clearly the size and shape of the areas requested, as well as the variables chosen, have major effects upon the accuracy and utility of the resulting data if confidentiality is not to be breached.

THE INITIAL, HARDWARE-BASED SOLUTION

Introduction

Our initial intention was to create a simple and easily changed prototype which would merely demonstrate the technical feasibility of the concepts. We intended to build this using a Fourth Generation Language for ease of modification: performance was not an important criterion at this stage. This was to have been carried out on university facilities and, in particular, on ICL equipment both to exploit the characteristics of the Content Addressable File Store (CAFS) and to replicate at least part of the long - standing OPCS/GRO(S) computing environment.

In the event, delays in the ESRC funding serendipitously proved helpful because ICL announced INDEPOL, a highly secure software system initially created for handling individual records for the defence sector and for the police. The advantages of INDEPOL over other, less hardware-specific solutions stem substantially from its exploitation of CAFS. In the first instance, the latter provides:

- fast searching at the disk. In principle, this permits searching at up to three megabytes per second on each disk and places little load on the central processor.
- multiple criteria searching: up to fourteen search criteria may be used simultaneously, with boolean or quorum logic using precise, stem, or fuzzy matching.
- the highest level of computer security yet available in the public domain. Access control by user, by terminal, by data name and data value, by use of passwords and other methods is claimed to ensure that, for instance, a particular 'high clearance' user can be prevented from performing functions which are legal in his own office whilst he or she is using an insecure terminal. The user cannot find out anything about data or facilities to which s/he does not have access: their very existence remains hidden.
- the ability to create menus and templates but also to make a query using a free format enquiry language. Macros may be shared and made generally available for frequent enquiries.

- ways for rules to be embedded in the system.

The reason for our selection of INDEPOL is not merely all of the above reasons, important as they are. In addition, it facilitates the move from the need for complete pre-planning of software to a situation where all the significant details of an application are held in the System Model and the Data Model: these can readily be edited - even within a single log-on - and, since the models are used interpretatively, there is minimum dependence on pre-compiled code.

The existence of INDEPOL - albeit at an early state of its product development cycle - therefore enabled us to revise our strategy. It enabled us to concentrate on building a Census - specific superstructure and, at the same time, to take advantage of the highly secure hardware and software facilities, including networked access, devised by ICL for their original clients. All of the work using INDEPOL was carried out in ICL's Defence Systems Division in Winnersh, Berkshire, UK.

Results

Our experience with the INDEPOL system is that it works extremely well for standard enquiries and simple presentation of either listed or singular results. The feasibility study has shown that individual queries are speedily answered, that tabular reports can be constructed and printed out on a peripheral device and that record-specific constraints can be applied to suppress or amend answers to queries. Moreover, standard facilities provide bit-compression of data fields wherever possible, minimising the data storage required. In essence, only two of the requirements of the system were not demonstrated satisfactorily : the addition of secondary geographical indexes (which, from seeing other INDEPOL applications, we are convinced would work well provided no more than about 10 indexes were required at any one time) and the public demonstration of the level of security (see below). More specific conclusions from the study include :

- (i) INDEPOL has a straightforward command language for querying the database; even complex cyclic control programming can be quickly mastered by the application designer.
- (ii) It is extremely secure and very versatile in the way in which applications, system and user security can be manipulated in terms of restrictions by data name, data value, command availability and option availability. In addition to controlling access, the system keeps complete records of all requests and commands. However, the sole availability of INDEPOL in a highly secure environment at the time of our tests, together with the need to work on Model 204, ensured that we could not carry out our initial intention - to throw open the system over the academic network with a challenge to colleagues to break into it!
- (iii) The CAFS system makes the INDEPOL database extremely fast (as fast as the disk control unit). Typically, a standard enquiry searching 1.2 million records and

cross-linking two files took rather less than 30 seconds. Many inquiries on a single file took only 2 to 4 seconds. A loop structure using cyclic control to construct tabular reports took less than 90 seconds to search 0.4 million records a total of 55 times and print the results.

- (iv) The design of the INDEPOL data model is straightforward in terms of the relationships between entities and attributes. The actual coding of the data model does require a certain amount of data processing expertise but can be quickly mastered by an experienced applications designer. From the user's point of view, how attributes relate to one another and to the physical world is evident from a brief study of the data model.
- (v) Template and menu design is very simple and there is an extensive validation ability which is always present in template construction. This allows for complex applications to be built up very quickly by even the most inexperienced of users. Use of the templates also permits querying of the database with only a limited knowledge of the INDEPOL command language.
- (vi) The ease of adding attributes post-hoc to the data model means that secondary features can readily be added to the system at a later date (e.g. the addition of either post-code or Ordnance Survey grid references to provide locational references) with the minimum detrimental effect on the performance of the system.

The majority of the problems encountered with INDEPOL stem directly from our use of the software for tasks for which it was not optimised or designed. The main problem areas are:

- (i) The production of two-dimensional tabular reports presents the major problem. On-screen production of tables is limited to 23 lines of text with no ability available to page through a large table (though it is possible to print the table through a spool file).
- (ii) In submitting tabular reports to the spool file, a large number of SCL and COBOL programs plus relatively complex and convoluted INDEPOL macros are required. An obvious means of reducing these problems would be to re-design the SAS tables into a series of standard layouts or choose a database system that combines the query ability of INDEPOL with a bolt-on report writing facility similar to RPT/RPF in ORACLE.
- (iii) There are only very rudimentary algebraic, and practically no statistical, manipulation (including sampling) capabilities in INDEPOL other than the use of standard operators and functions such as SUM and MEAN. It is envisaged that a 1989 release of INDEPOL will go some way towards catering for these particular requirements.

THE SOFTWARE - BASED SOLUTION

Work involving Model 204 is, at the time of writing, at a much earlier stage of development than that of INDEPOL. Initial familiarisation with the MVS/XA operating system running on the Amdahl 5890-300E computer and with the Model 204 software (which is currently used only by the Librarians in Manchester), together with the need to re-load and index the test data, has ensured that initial demonstrations of a primitive system could not be given until end-January 1989.

Nonetheless, some preliminary conclusions can be made though it must be appreciated that these may reflect inexperience with Model 204's capabilities. These are as follows :

- (i) Model 204 is highly unlikely to provide as secure a system as INDEPOL or any system exploiting CAFS.
- (ii) Similarly, it is unlikely to provide as rapid response to simple queries as INDEPOL unless much indexing has been carried out; such indexing appears extremely expensive on storage space. INDEPOL scores heavily in situations where unanticipated queries are made.
- (iii) Model 204 is however relatively easy to use, has SQL and has the benefit of good report generation capabilities. For production of complex standard tables, it may well be more efficient in terms of both human and computer resources than INDEPOL.

CONCLUSIONS

All of our work is predicated upon the notion that the data base is run within the Census offices and accessed by one means or another. Given this, the relative merits of the two solutions seem likely to be exploitable in different modes of operation. Thus it seems likely that technical considerations of an on-line querying system could be satisfied by INDEPOL; this of course does not address directly the question of public perception of the risk of disclosure. Based upon our experience to date, the following scenarios might be feasible ones :

- (i) Use of INDEPOL (or, at least, CAFS-based systems - CAFS units are now available on SUN systems) to provide an on-line enquiry service over national telecomms networks. Conceptually, this would be simply an extension of the highly successful National On-line Manpower Information System (Nelson and Blakemore 1986). It would be viewed as a parallel facility to the distribution of conventional statistics for standard areas (see below).

To make this work, groups of users would have to be authorised in the same way in which over 200 existing NOMIS groups have been authenticated. These include central and local government, the commercial sector agencies and academics. All who do not have 'as of right' access (as local government does to some datasets held on NOMIS) would require licensing.

Users would therefore access the 'raw' questionnaire responses but would only be permitted to receive aggregate output, suitably screened to ensure non-disclosure. We are quite confident that this could be achieved by connecting the CAFS machine directly to the network though constraints exist in terms of the type of terminal needed if INDEPOL is used. Turn-around of the order of seconds or, at worst, minutes would be available by this approach.

- (ii) Use of Model 204 to run a service based upon OPCS receiving queries in one of a number of forms; the best solution would be for users to compile queries and dispatch them over telecomms to OPCS who then transfer these to a non-networked machine on which the database resides and produce the results required. Depending on the mode of transfer, this will provide responses in the time scale of an hour to several days.

Both of these scenarios would require the appointment of a data certification officer who would be responsible for monitoring the flow of requests. Both would benefit from the construction and low-cost distribution of a Query Formulator for PCs which would enable users to design their requirements in a way suitable for the system and for transmission to OPCS, with obvious syntax and logic errors being weeded out at source. Equally, both could have advantages for OPCS and GRO(S) not yet mentioned : they would diminish the penalties for failing to anticipate user demand correctly, they might facilitate the construction of new products such as Public Use Samples and they would simplify the design of samples for other surveys and data linkage for files held within the remit of the Registrar Generals (e.g. demographic, housing and mortality and morbidity data).

Nothing of what is proposed is totally novel. Credit rating agencies (e.g. TRW in the USA) already operate larger data bases and provide country - wide access to the data. An OPCS - fostered scheme, the Longitudinal Study, already gives a form of on-line access in the UK to anonymised records of sampled individuals. Moreover, this particular proposal has to be seen in the context of many other developments in data collection in Britain, notably the compilation of the Community Charge register by local governments (Redfern 1988). What is critical, however, is the paramount need to ensure that no disclosure of information about identifiable individuals occurs.

The costs of setting up and running such a system are not yet quantifiable in any detail, in part because of reviews of charging mechanisms and the likely introduction of new facilities in the census offices. Assessment of costs and benefits will also depend upon decisions of government on financial recovery targets from production of the census data; as one example, the lowest effort and cost method of distributing standard statistics would be to produce one CD-ROM covering the whole country and sell this to every customer, even if they only required data for one small area (as is generally the case).

By its very nature as a previously unavailable service, the level of customer interest is unquantifiable in any detail. However, in a survey of the academic community (Marsh et al, 1988) no less than 76% of respondents said they would use such a tool (subject to its cost) and a further 22% said they were undecided. The NOMIS experience suggests that, with appropriate training and publicity, a wide variety of users can be expected; for many of them, doing without NOMIS would now seem inconceivable. Finally, availability and success of such a system could have implications for the commercial census agencies licensed by OPCS after 1981; as well as continuing as purveyors of standard statistics to naive or very infrequent users, they would have to become census skills centres and act as intermediaries for any customers who required help in formulating queries to the proposed system.

ACKNOWLEDGEMENTS

The developments described in this report would not have been possible without the help and advice of a number of other individuals. Thanks must first go to Martin Higgins, now of Pinpoint Ltd, for the initial development work. In addition, Mark Musto and Geoff Sprott (part of the ICL INDEPOL Support Team), Dick Morgan (ICL INDEPOL Applications Manager), and also Miriam Glyde and Dick Goodwin (of ICL Winnersh) are owed heartfelt thanks for their support and advice in the earlier stages of development. Colin Wymer kindly made a copy of the data used in the study. In OPCS and GRO(S), Chris Denham, Basil Mahon, David Pearce, Terry Russell and Frank Thomas have been invariably helpful in all matters related to this project. Keith Cole and colleagues in the University of Manchester Regional Computer Centre have been immensely helpful in regard to our use of Model 204.

REFERENCES

- Bethlehem J.G., W.J. Keller and J. Pannekoek (1988) Disclosure control of microdata. Proc.4th US Annual Research Conference of Statisticians, 181-92.
- Carmichael, J.W.S. (1987) INDEPOL : A software package exploiting CAFS. ICL Defence Technology Centre, Winnersh
- Denham C.J. and Rhind D.W. (1983) The 1981 Census and its results. in Rhind (1983), 17-88.
- Dewdney J.C. and Rhind D.W. (1986) The British and USA's Censuses of Population. in M. Pacione (ed.) Population Geography : Progress and Prospect, 35-57, Croom Helm, London
- DoE (1987) Handling Geographic Information. The report of the Committee of Enquiry chaired by Lord Chorley. HMSO, London.
- Duncan G.T. and Lambert D. (1986) Disclosure-limited data dissemination. J. Amer. Statist. Assocn. 81, 393, 10-28.
- Duncan G.T. and Lambert D. (forthcoming) The risk of disclosure for Microdata. J. of Business and Economic Statistics
- HMSO (1988) 1991 Census of Population, Command Paper 430
- Marsh, C., Arber S., Wrigley N., Rhind D. and Bulmer M.(1988) The View of Academic Social Scientists on the 1991 Census. Environment and Planning A, 20, 851-89.
- Nelson R. and Blakemore M.J. (1986) NOMIS - a national GIS for the management and mapping of employment, unemployment and population data. Tech. Papers 1986 ACSM/ASPRS Annual Convention, Vol. 1, 20-30.
- OPCS (1988) 1991 Local Statistics : a discussion paper, Office of Population Censuses and Surveys, London.
- Redfern P. (1987) A study on the future of the census of population: alternative approaches. Eurostat Theme 3, Ser. C. Luxembourg : Office of Official Publications of the European Community.
- Redfern P. (1988) Population registers : some administrative and statistical pros and cons. J. R. Statist. Soc. A 151, 3.
- Rhind D.W. (1983) ed. A Census User's Handbook, Methuen, London.
- Rhind D.W. (1984) The SASPAC story. BURISA, 60, 8.
- Rhind D.W. (1985) Successors to the Population Census. J1. Economic and Social Measurement, 13, 29-42.
- Wise S. and Pellatt R. (1988) Rapid access to a large cartographic data base using CAFS : a feasibility study. Tech. Rept. 4 in GIS, Computing and Cartography, Univ. Bath

A CARTOGRAPHIC EXTRACT OF THE TIGER FILE: IMPLICATIONS FOR MAPPING APPLICATIONS

Amy Bishton
Geography Division
U. S. Bureau of the Census
Washington, DC 20233

ABSTRACT

The Topologically Integrated Geographic Encoding and Referencing (TIGER) File is the Census Bureau's primary data base for map production. Collecting and chaining the TIGER File's topological elements of 0-cells (points), 1-cells (lines) and 2-cells (areas) into meaningful cartographic elements is a major step in map creation. Rather than have each mapping application directly access the TIGER File and independently perform the chaining process, the Census Bureau creates a secondary data base -- a cartographic extract from which mapping applications retrieve the coordinate strings of pre-chained cartographic elements. While developed initially to reduce overall computer processing requirements, the cartographic extract has additional implications for mapping applications, such as providing a structure for approaching the mapping task and enhancing preproduction decision-making processes.

INTRODUCTION

The Topologically Integrated Geographic Encoding and Referencing (TIGER) System represents the culmination of efforts by the United States Bureau of the Census to automate the geographic support processes for the 1990 decennial census and beyond. A major component of these geographic support processes is the production of maps, initially to aid data collection and later to complement data dissemination. The Census Bureau has produced a variety of map types from preliminary versions of the TIGER File to support pre-1990 test census applications. Additionally, the Census Bureau currently is producing maps from the "live" TIGER File in preparation for the 1990 decennial census.

The TIGER File breaks down the network of roads, railroads, hydrography, political boundaries and other features found on the earth's surface into a system of 0-cells (points), 1-cells (lines), and 2-cells (areas). Latitude and longitude coordinate values for 0-cells and 1-cells are stored explicitly, while coordinate values for 2-cells are derivable from the 0-cells and 1-cells. Attribute information about these topological elements, such as the feature name of a 1-cell or the geographic cover of a 2-cell, is stored in numerous related subfiles (Kinnear, 1987).

Collecting and chaining the TIGER File's topological elements into meaningful cartographic elements based on attribute information is a major step in map creation. Furthermore, many mapping applications

require the retrieval and chaining of topological elements in the same or a similar way. Rather than have each mapping application directly access the TIGER File and independently perform the chaining process, the Census Bureau creates a secondary data base, the cartographic extract from which mapping applications retrieve the coordinate strings of pre-chained cartographic elements (Bishton, 1988). While initially developed to reduce overall computer processing requirements, the cartographic extract has additional implications for mapping applications.

APPROACHING THE MAPPING TASK

Perhaps most significantly, the cartographic extract provides a structure for approaching the mapping task not readily offered by the topologically-based TIGER File. First, the extract's data base structure suggests a conceptual framework that is consistent with traditional mapping methods. Second, the extract's content supports specific mapping strategies that are compatible with an automated environment.

A Conceptual Framework

The cartographic extract consists of logical subfiles linked by a system of pointers (see Figure 1). Each subfile contains fixed-length records, with each record holding pointer data followed by nonpointer or descriptive data. Balanced tree (B-tree) logical subfiles, also referred to as directories, store records ordered on a key; random access logical subfiles (RALS) store records randomly.

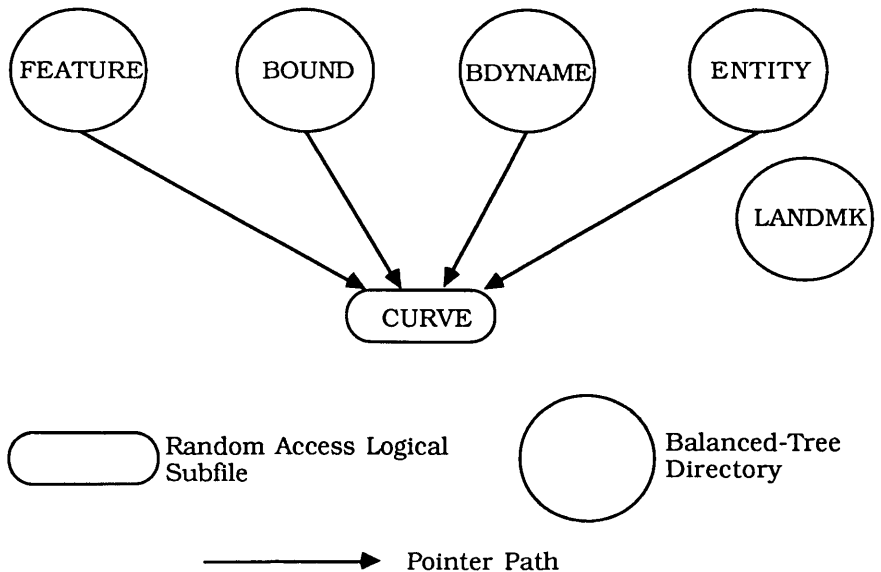


Figure 1. Cartographic Extract Data Base Structure

The FEATURE directory holds records for linear and areal features, including physical features such as streams, glaciers, and ridge lines, and cultural features such as roads, railroads, and airports. The LANDMK directory holds records for physical and cultural point features such as mountain peaks, lookout towers, and churches. Together, those two directories embody the concept of a feature base and a feature names overlay.

The BOUND directory holds records for unique combinations of political and statistical boundary coincidence; for example linear chains that function simultaneously as a county boundary and an American Indian reservation boundary represent a unique boundary combination that is different from linear chains that function simultaneously as a county boundary and a city boundary. Furthermore, because the individual identity of each geographic entity has no bearing on the boundary combination, a linear chain that forms the boundary of county A and county B is not uniquely different from a linear chain that forms the boundary of county A and county C. As the source for boundary symbology, the BOUND directory equates with the concept of a political base.

A political names overlay may be thought of as residing in the BDYNAME and ENTITY directories. The BDYNAME directory allows the linear placement of names parallel with boundary symbology; the ENTITY directory allows the areal placement of names centered within a specific geographic entity. The BDYNAME directory holds records for unique combinations of opposing entity names along state and county boundaries. Unlike the BOUND directory, the individual identity of each geographic entity has a bearing on the boundary name combination; consequently a linear chain that forms the boundary of county A and county B is uniquely different from a linear chain that forms the boundary of county A and county C. The ENTITY directory holds records for most of the political and statistical geographic entities recognized by the Census Bureau for data collection and data tabulation purposes. A record exists for each discontinuous piece of an entity so that every geographic area can be labelled individually.

The map base and map names overlay concepts cannot be fully realized without the CURVE RALS. Each record in the FEATURE, BOUND, BDYNAME, and ENTITY directories has a pointer to a series of records in the CURVE RALS to link a cartographic data element with its chained coordinates. A LANDMK directory record does not require a link to the CURVE RALS because its cartographic data element possesses only one coordinate point, which is stored directly on the LANDMK directory record.

Mapping Strategies

The record content of individual subfiles allows a progression from a conceptual framework toward specific strategies for approaching the mapping task. Figure 2 shows the record layout for each subfile, excluding pointer data. Shading identifies the fields that make up a directory's key. Some noncritical fields have been omitted and others rearranged for the purposes of discussion.

FEATURE DIRECTORY

COORDINATE ENVELOPE	FEATURE CODE	FEATURE NAME	CHAIN NUMBER	ENDPOINT COORDINATE	ADJACENT PARTITION	BOUNDARY FLAGS
---------------------	--------------	--------------	--------------	---------------------	--------------------	----------------

BOUND DIRECTORY

COORDINATE ENVELOPE	BOUNDARY CODE	CHAIN NUMBER	ENDPOINT COORDINATE	ADJACENT PARTITION
---------------------	---------------	--------------	---------------------	--------------------

BDYNAME DIRECTORY

COORDINATE ENVELOPE	ENTITY CODE RIGHT	ENTITY CODE LEFT	ENDPOINT COORDINATE	ADJACENT PARTITION	ENTITY NAME RIGHT	ENTITY NAME LEFT
---------------------	-------------------	------------------	---------------------	--------------------	-------------------	------------------

ENTITY DIRECTORY

COORDINATE ENVELOPE	ENTITY CODE	CHAIN NUMBER	ENDPOINT COORDINATE	ADJACENT PARTITION	ENTITY NAME
---------------------	-------------	--------------	---------------------	--------------------	-------------

CURVE RALS

COORD	COORD	COORD	COORD	COORD	COORD	COORD	COORD
1	2	3	4	5	6	7	10

LANDMK DIRECTORY

POINT COORD	FEATURE CODE	FEATURE NAME
-------------	--------------	--------------

Figure 2. Cartographic Extract Subfile Record Layouts

The coordinate envelope field describes the spatial extent of a chain. The field contains four subfields: the chain's minimum longitude, maximum longitude, minimum latitude and maximum latitude. As the first field of a directory's key, the coordinate envelope is the primary data element by which the directory orders individual records. Directory records are ordered spatially to allow mapping applications to determine quickly which chains fall within a rectangular area to be mapped. While the TIGER File is conceptually a single computer file, it actually consists of many physical files or partitions, with each partition usually covering the area of one county. Because an extract file exists for each TIGER File partition, the cartographic extract also is county-based. For large-scale mapping applications, the area to be mapped often represents a small subset of the county covered by an extract file. By ordering records spatially, many chains falling outside the mapping area may be skipped without a direct comparison between the mapping area's coordinate envelope and each chain's coordinate values.

Just as the coordinate envelope provides a spatial filter within an extract file, the adjacent partition field provides a spatial filter across extract files. If the mapping area extends beyond a county, the mapping application must access more than one extract file to produce a single map. A cartographic problem arises, since each 1-cell along a partition edge is found in two TIGER File partitions and the 1-cell coordinates are incorporated into chains in two extract files. To avoid overprinting symbology, the mapping application must access a chain from only one extract. The adjacent partition field indicates if a chain occurs on a partition edge, and contains the code of the adjacent partition. Following preset rules, such as accessing only from the lower-code partition, the mapping application selects or rejects chains falling on the partition edge.

The feature code and boundary code fields provide potential for cartographic classification and generalization. Turning to the feature code first, every 1-cell and some 0- and 2-cells in the TIGER File carry a Census Feature Class Code (CFCC) (Trainor, 1986). A feature code represents a predefined grouping of CFCCs that can be considered equivalent for mapping purposes; for example, feature code 1 combines 10 CFCCs all under the general heading of "Interstate, U.S., and State highway, not in tunnel or underpassing." If the extract feature groupings are still too numerous for a specific mapping application, groupings can be combined by changing the feature code of several similar groupings to the feature code of one representative grouping. By referencing application-specific lookup tables to direct the feature code changes, one generic mapping program easily produces a variety of map designs.

The boundary code for a chain represents its unique combination of boundary coincidence. The code is the integer equivalent of a boundary number to the 21st power where the nth bit is "on" if the nth boundary type is present; for example, the boundary code of a chain forming a state boundary (bit 20) and a county boundary (bit 19) but no other boundary type equals 786,432 ($2^{20} + 2^{19}$). No mapping application ever requires all 21 boundary types to be symbolized, and some mapping applications require two or more boundary types such

as incorporated place boundaries and census designated place (CDP) boundaries to be symbolized as the same boundary type. The bit structure of the boundary code easily accommodates this cartographic generalization and classification. The user redefines the boundary code by turning off all bits of boundary types not required, and by turning on the bit of a representative boundary type (e.g., incorporated place) while simultaneously turning off the bit of any other boundary type it represents (e.g., CDP). As with the feature code changes, a lookup table directs boundary code changes.

Simply altering a feature code or boundary code does not automatically trigger the merging of similar chains into a single, larger coordinate chain. If a mapping application symbolizes all railroad-related feature codes identically, chaining together all like-named chains within the appropriate range of feature codes will maximize the regularity of the railroad symbol pattern and minimize the repetition of the railroad name. If the same mapping application symbolizes only 5 of the possible 21 boundary types, chaining together all extract chains with the same newly-defined boundary code promises a more regular boundary symbology pattern. The mapping application accomplishes this secondary chaining by first identifying which chains belong to a common group using a work directory, and then matching chains within a group on their endpoint coordinates.

Frequently, Census Bureau mapping applications call for certain feature types, such as pipelines and powerlines, to be shown only where coincident with a mapped boundary. Including all pipelines and powerlines leads to unnecessary clutter, while showing no pipelines or powerlines may create confusion at the mapped boundary. The boundary flag field found on the FEATURE directory record provides a connection between feature and boundary chains. Within the boundary flag field, the nth bit is turned on if the nth boundary type is coincident anywhere along the feature chain. By checking whether or not certain boundary bits are turned on, the mapping application selectively suppresses feature chains within a feature group.

The entity code field provides a means of merging statistical values from external files with the extract's cartographic base to produce choropleth maps. The entity code is a character string where the first alphabetic character represents the geographic entity type (e.g., C=county) and the remaining numeric characters identify a specific geographic entity (e.g., 20005=Atchison County, KS). The entity code adheres to the Census Bureau's highly developed and extensively recognized geographic code scheme whereby each geographic entity receives a unique code incorporating hierarchical relationships between entity types (U.S. Census Bureau, 1983). Given the Census Bureau's current focus on data collection activities rather than data dissemination activities, no choropleth maps have been produced from a cartographic extract, although the potential remains.

ENHANCING PREPRODUCTION PROCESSES

Prior to map production, the mapping application selects an appropriate map scale and determines the grid sheet layout including

any insets. Increasingly, mapping applications use the cartographic extract to enhance these preproduction design-making processes. In particular, the accessibility of the coordinate envelope or the full coordinate chain of a geographic entity from the extract's ENTITY directory represents an invaluable time-savings in preproduction processing.

The County Block Map, a developing map series designed to clearly display each census block within a county usually requires multiple map sheets at a single scale to cover one county. The map scale is determined by a process that measures overall feature density via the TIGER File (Martinez, 1987). The grid sheet layout for the county is then a function of map sheet size and map scale. When the grid sheet layout approximates the county subdivision boundary network depicted on the map, slivers of county subdivisions appear along map edges. Slight shifting of the grid sheet layout to maximize the number of county subdivisions that fall on a single sheet reduces the slivering effect. After comparing the map sheet coordinate envelopes against the county subdivision coordinate envelopes located in the ENTITY directory, the number of single-sheet county subdivisions is summed for nine different shift directions: a shift to the north, to the south, to the east, to the west, to the northeast, to the northwest, to the southeast, to the southwest and no shift at all. The shift direction producing the largest number of single-sheet county subdivisions is applied to the grid sheet layout.

Insets for a County Block Map are identified by a process that locates local pockets of high feature density via the TIGER File (Martinez, 1989). Since high feature density typically occurs within urban areas, many insets encompass incorporated places and CDPs. When a rectangular inset only approximates the irregular limits of a place, slivers of a place remain outside the inset area. By comparing the inset coordinate envelopes to the place coordinate envelopes located in the ENTITY directory, insets are expanded selectively to reduce the clutter of place slivers.

Another map series, the County Locator Map, uses the extract exclusively for all preproduction processes. The County Locator Map shows the boundary of and centers a label within each address register area (ARA) (a 1990 census collection unit) within a county while showing few cultural or physical features. The preproduction process attempts to strike a balance between maximizing the map scale for label legibility and minimizing the number of map sheets and inset areas for ease of use. The ARA coordinate chains are scaled and ARA label placement is tested through several iterations of grid sheet layouts, progressing from the simplest grid sheet layout (that is, 1 x 1 sheet) to more complex grid sheet layouts (for example, 2 x 3 sheets). Groups of ARAs that cannot be legibly labelled within an iteration form inset areas. The iterative process stops when a more complex grid sheet layout (in effect, increasing the map scale) does not reduce the number of inset areas significantly.

CONCLUSION

As a growing number of mapping applications accesses the cartographic extract, the initial investment in computer processing time to create the extract is returned many times over. Moreover, new and unplanned uses of the extract can initiate modifications to the original extract concept, resulting in a more robust cartographic data base. Modifications currently in development include storing the left-side and right-side geocodes for each linear chain in the FEATURE, BOUND and BDYNAME directories. The left/right geographic information can be used to selectively suppress linear chains falling within specified geographic entities. As we move toward the 1990 decennial census, the cartographic extract concept will continue to evolve to meet the Census Bureau's mapping needs.

REFERENCES

Bishton, A. 1988, Designing and Using a Cartographic Extract: Mapping from the TIGER System: Proceedings URISA, Vol. II, pp. 130-141.

Kinnear, C. 1987, The TIGER Structure: Proceedings, Auto Carto 8, pp. 249-257.

Martinez, A. 1987, Applications of Expert Rules in Automated Cartography, presented at Applied Geography Conference.

Martinez, A. 1989, Automated Insetting: An Expert Component Embedded in the Census Bureau's Map Production System: Proceedings, Auto Carto 9.

Trainor, T.F. 1986, Attribute Coding Scheme: Identification of Features in the U.S. Census Bureau's TIGER System: Proceedings, Auto Carto London, Vol. I, pp. 117-126.

U.S. Bureau of the Census, 1983, 1980 Census of Population and Housing, Geographic Identification Code Scheme, U.S. Government Printing Office, Washington, DC.

A VERSATILE MAPPING SYSTEM FOR THE USGS 1:100,000 DLGs

David J. Cowen
and
Timothy R. White
University of South Carolina
Columbia, S.C. 29208

ABSTRACT

This paper describes a versatile mapping system for handling the USGS 1:100,000 digital line graphs. It was developed to automatically generate maps of any scale for any part of South Carolina. The system is based on a set of FORTRAN procedures that access and manipulate any number and combination of DLG files. Using a command language structure the procedures allow the user to select attributes, set windows, change scale, shade polygons, and assign line symbols and shading patterns. The system automatically combines the necessary files into panels for a study area and generates the legend and scale. The system is being used to generate a wide range of maps for a GIS that is designed to analyze state wide infrastructure needs and industrial site selection. For example, it has proven to be an excellent system for creating a diverse set of base maps that are used by local governments to compile additional coverages such as sewer and water lines.

INTRODUCTION

The 1:100,000 Digital Line Graphs

The creation of the 1:100,000 digital Line Graphs (DLG) represents a milestone in the development of a national digital cartographic database. As a result of the cooperative program between the United States Geological Survey (USGS) and the Bureau of the Census there now exists a digital representation of the transportation and hydrographic features of the United States (Callahan and Broome 1984). Throughout the United States organizations at every level are experimenting with the data for a wide range of applications.

For many users the 1:100,000 DLG data are somewhat of a mixed blessing. Although they provide an inexpensive existing digital cartographic base file they do not fulfill all of the user needs in terms of scale, accuracy, content or coverage. In practice, even though the data only meet accuracy standards at 1:100,000 the Bureau of the Census and numerous other organizations have "cheated" and enlarged them even beyond 1:24,000 for applications not requiring positional accuracy. Furthermore, even though the more than 200 DLG attribute codes represent a fairly complete set of the hydrographic, highway, railroad and miscellaneous transportation features found on standard USGS quadrangles they only include a few feature names. Additional problems arise when the files need to be converted into continuous data bases required for geographical information systems.

It should be noted that problems with the 1:100,000 DLG base are symptomatic of the current status of digital cartographic data and, more importantly, maps in general. Through the combined efforts of the USGS and the Bureau of the Census there now exists an extremely valuable topologically structured data base. These files provide a clear indication of the type of information that will be contained in the National Digital Cartographic Database. As such they provide an excellent basis for any organization to start to plan for future applications of digital cartography and GIS.

Importance to South Carolina

The DLG data provided a basis for the creation of a truly state-wide GIS that would help analyze the needs in South Carolina for infrastructure improvements and economic development decisions. Before the DLG base was created most research projects were forced to compromise with respect to either spatial extent or spatial resolution. The DLG data represent an excellent compromise for a state the size of South Carolina. It provides an unbiased and uniform coverage of transportation and hydrography throughout the state that will serve as a useful basemap until the higher resolution 1:24,000 scale data becomes available in the next decade.

The Need for a Mapping System

Once the DLG files arrived in the Fall of 1987 there was an urgent need to generate a series of output products. Initially, it was important to demonstrate that the data would be appropriate for state wide GIS applications. This "proof of concept" stage of the project required a versatile mapping system that could quickly generate high quality map products at a wide variety of scales with any combination of attributes for any section of the state. After considerable discussion and experimentation with the files it was evident that it would be desirable to have a mapping system that met the following criteria:

1. The user must be able to select any size rectangular window anywhere in state
2. It should be possible to specify the window in Latitude and longitude, UTM coordinates or with a center point and scale
3. The user must be able to select any combination of DLG attributes
4. The user must be able to generate maps at any scale
5. The user must be able to assign variable shading and line patterns to area and linear features
6. The program must automatically generate a legend that displays the attributes and related symbology
7. The user must be able to overlay grids in either latitude and longitude or UTM coordinates
8. It must be possible to export the selected data to other programs such as GIMMS, AUTOCAD, and SAS
9. The system should have a simple command language interface with a reasonable set of default settings

IMPLEMENTATION

Hardware Considerations

The first part of system implementation involved the selection of the hardware and software environments. Although Luman (1987) and others have developed personal computer based programs that merge and display the DLG files the volume of data for a state wide system prohibited such an approach. For example, even the small state of South Carolina consists of 575 7.5 minute quadrangles or 1150 individual data sets for the transportation and hydrography information. When combined with the 30 minute files for miscellaneous transportation the total data base consists of 1238 files that consume over 2 gigabytes of storage. It was obvious that the system should reside on the University main frame network where it could utilize disk rather than tape storage. It is also important to note that the system was designed to generate plots on a 24 inch electrostatic plotter with a resolution of 400 dots per inch and its own internal rasterization system. This type of output device is capable of quickly generating high quality output without operator intervention.

Software Considerations

It was also quickly determined that no existing software system would be able to meet the stated requirements. Since the program was designed only as a mapping system the individual 7.5 or 30 minute DLG data sets could be handled as needed without the need to construct continuous spatial databases required by polygon based mapping systems or GISs. For example, if a large water body extends over several quadrangles only the topology within each quadrangle is important for display purposes and the entire spatial entity such as a lake does not need to be created. Therefore, the system was designed to take advantage of the UTM coordinates and existing topology of the DLG optional format rather than incurring the extensive overhead required to create and store a fully integrated data base. Implementation within a CAD structure was dismissed for the same reasons plus the difficulty of handling multiple attributes in that environment. Therefore, the decision was made to create a special purpose mapping system that had limited functions but responded quickly to a variety of mapping needs.

The Driver Program

The final system consists of a driver program and plotting program. The driver program is the file management part of the system. It determines which data sets contain the DLG attributes for the window that has been selected. This part of the process involves converting the selected window into decimal degrees and searching a catalog of the data sets. The driver program automatically generates the required number of panels, proper job control data definition statements and submits the job. In fact, if the user is generating a series of the same type of maps for numerous windows throughout the state the driver program will automatically submit the proper number of jobs and calculate an estimated execution time. Furthermore, with this driver program it would be possible to map the entire state at any scale by simply specifying a bounding window. In practice this front end has facilitated

the rapid generation of literally hundreds of maps with no operator intervention.

The Plotting System

Once the proper files have been defined by the driver program the plotting program utilizes the coordinates of the window and the scale to clip the data and determine the size of the resultant map. In practice, the system is typically used with 24 inch paper and scales are often adjusted to fit within that dimension. The actual plotting system consists of more than 60 FORTRAN subroutines than make extensive use of the CALCOMP plot library. The user interface to the system is based on a series of card images that contain parameters for the following set of commands:

ALLATT - SELECTS ALL ATTRIBUTES FOR PLOTTING AND / OR
EXTRACTION. (Figs. 1 and 2)

ATTRIB - ATTRIBUTE CARDS.

CNTRPT - ALLOWS USER TO SELECT A CENTER POINT AND MAP
DIMENSIONS IN INCHES TO DEFINE MAP WINDOW INSTEAD
OF THE WINDOW

FILE - TURNS DATA FILE EXTRACTION FOR ARCS AND POLYGONS ON
OR OFF.

FILES - SET FILES STARTING POSITION AND NUMBER OF FILES TO
BE CONSIDERED FOR PROCESSING.

FREQ - GENERATE ATTRIBUTE FREQUENCY LIST.

GRID - TURN UTM GRID GENERATION ON OR OFF. (Fig. 3)

LABSCL - DRAW MAP SCALE LINE?

LEGEND - TURN LEGEND FOR PLOT ON OR OFF.

OTHERD - IMPORT AND PLOT ANOTHER FORMAT OF DATA?

PCOPY - SET NUMBER OF PLOTS TO BE GENERATED.

PLOT - PLOT GENERATION ON OR OFF.

PRZONE - UNIVERSAL TRANSVERSE MERCATOR MAP PROJECTION ZONE.
USED BY THE GENERAL CARTOGRAPHIC TRANSFORMATION
PACKAGE V1.0 TO CONVERT DECIMAL DEGREES TO METERS.

SCALE - SET PLOT SCALE.

TITLE - SET PLOT TITLES.

TITLBOX - DEFINE A BOX FOR USER DEFINED TITLES.

VERBOS - SETS LEVEL OF PROGRAM MESSAGES.

WINDOW - DEFINE WINDOW OF DLG DATA TO BE EXTRACTED WITH
LAT/LONGS OR UTM COORDINATES.

With the set of commands it is possible to generate an infinite set of maps such as the composite map of North Charleston (Fig. 4).

APPLICATIONS

During the course of the development of the system it was important to generate a series of base maps that clearly demonstrated the content of the DLG files and their applicability for state wide GIS applications. A map of the southeastern part of the state (Fig. 5) was one of the first maps generated. This one map was extremely instrumental in proving that the concept would work and helped establish the credibility of the research team. The system has been used to generate scores of such maps usually with only a couple of hours turnaround.

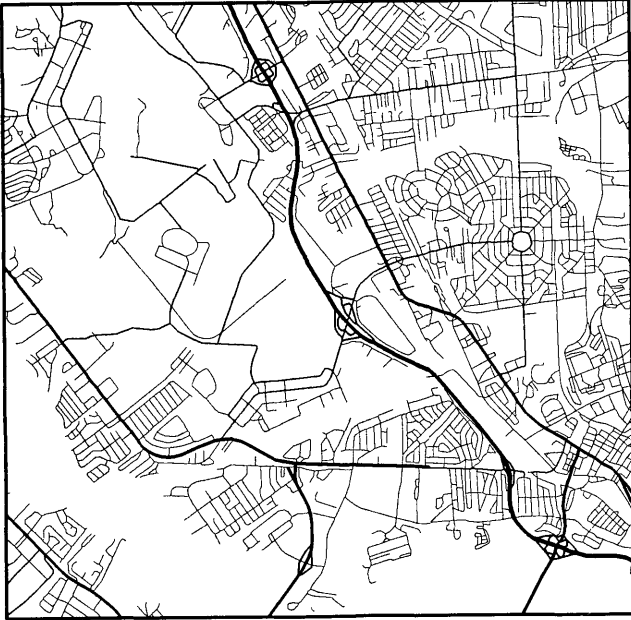


Fig. 1 Highway and Road Attributes - Charleston, S.C.
Scale 1:80,000



Fig. 2 Railroads, Sidings and Stations - Charleston, S.C.
Study Area

590000mE

596470. . 3641766.

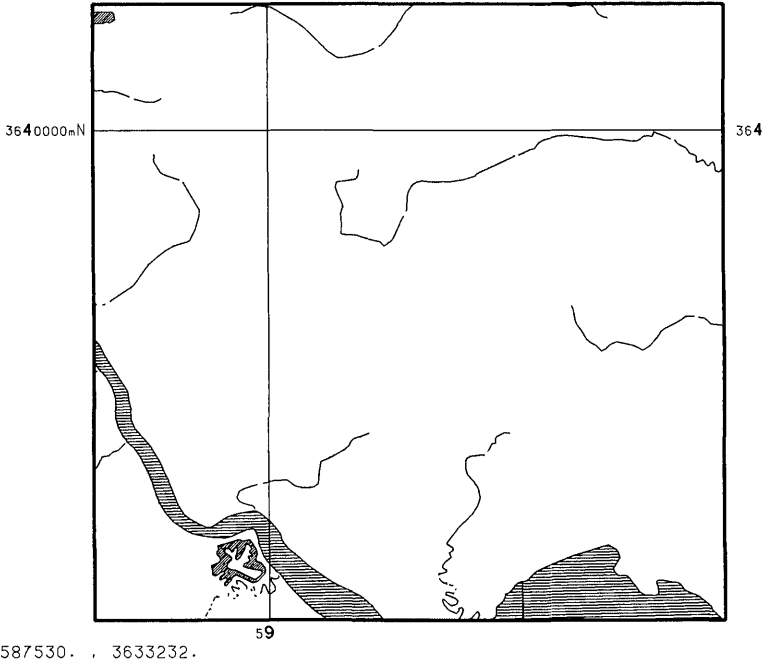


Fig. 3 Hydrographic Attributes With UTM Grid

GIS Applications

The second major use of the system has involved the creation of base maps of sewer and water districts. As a major part of the economic development project the ten regional planning councils in the state have been hired to create maps of the sewer and water lines that are relevant to economic development decisions. In order to assure that the maps can be registered to ARC/INFO transportation coverages it was important to provide the planners with maps that were derived from the DLG base. Using the mapping system it was possible to easily generate a base map of the existing transportation and hydrography for each of the 575 water districts throughout the state. Although the final maps were plotted at scales of 1:20,000 or 1:15,000 the system was also used to generate a series of 4 by 6 inch reference maps (Figs. 6 and 7). These proof maps were generated simply by specifying a center point for the water district and a scale. The planners are now transferring the water and sewer lines onto the full scale base maps and the data are subsequently being added to the statewide GIS simply by selecting the relevant arcs from the transportation layer within the GIS software.

EVALUATION

Within a few months the system has proven to be an important adjunct to the overall geographical data processing capabilities of the University of South Carolina. The system

596470. , 3641766.

59000mE

00'

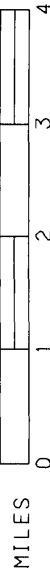


3640000mN

59

587530. , 3633232.

80°03'53" , 32°50'07"



MAP SCALE IS 1:80000.

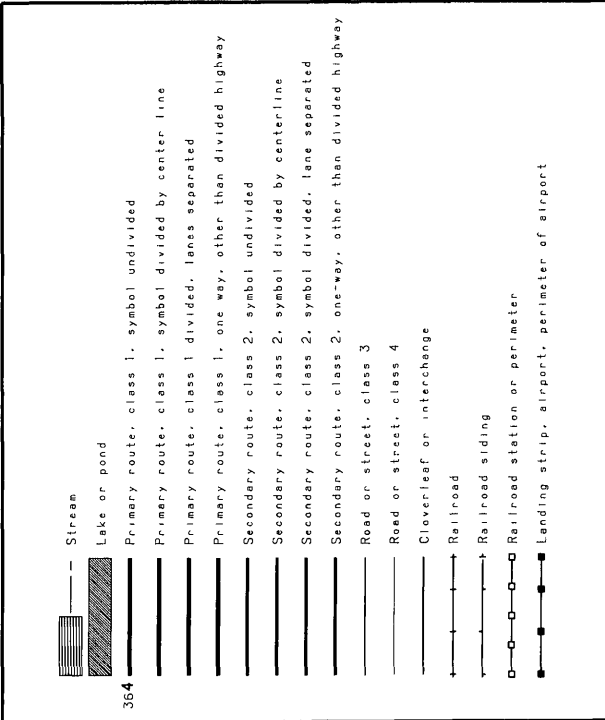


Fig. 4 Composite Map of Study Area - with Legend, Scale and Grids.

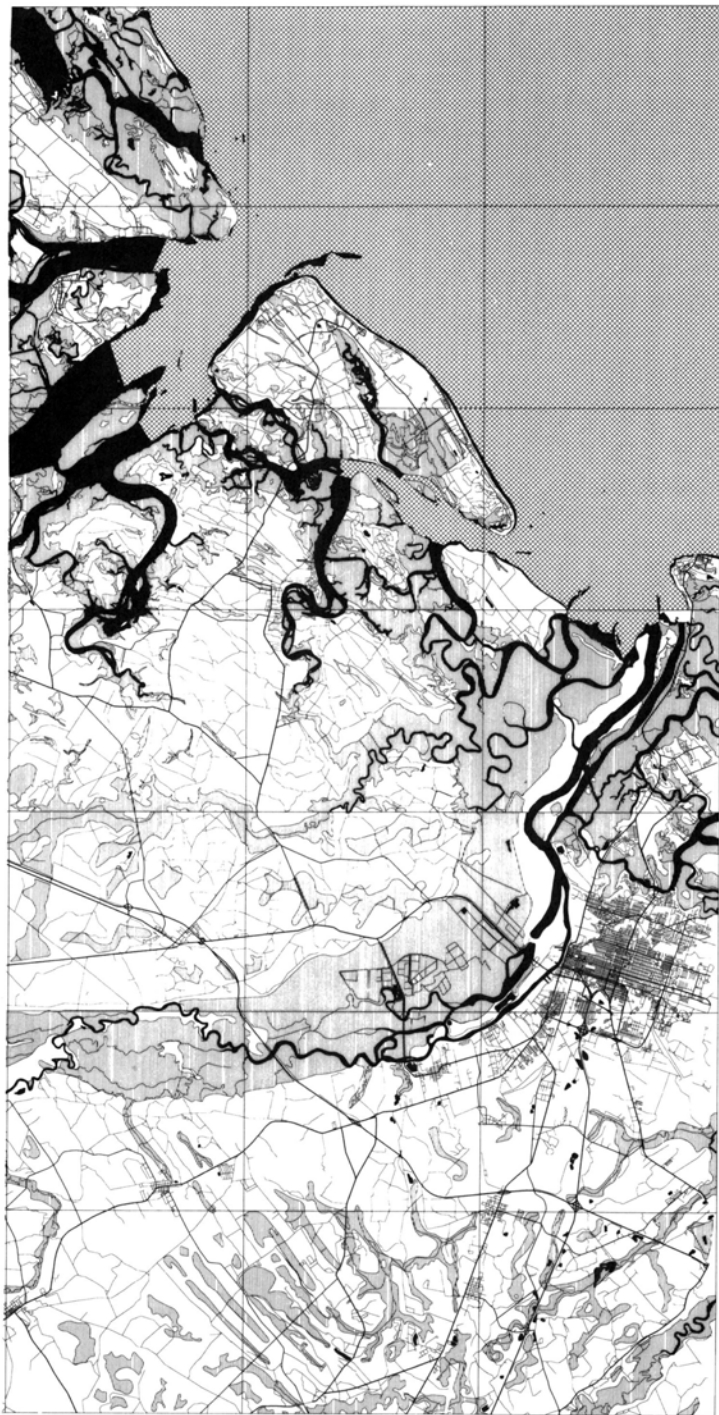


Fig. 5 Hilton Head Island and Savannah Georgia - Composite Panel consisting of 21 quadrangles.

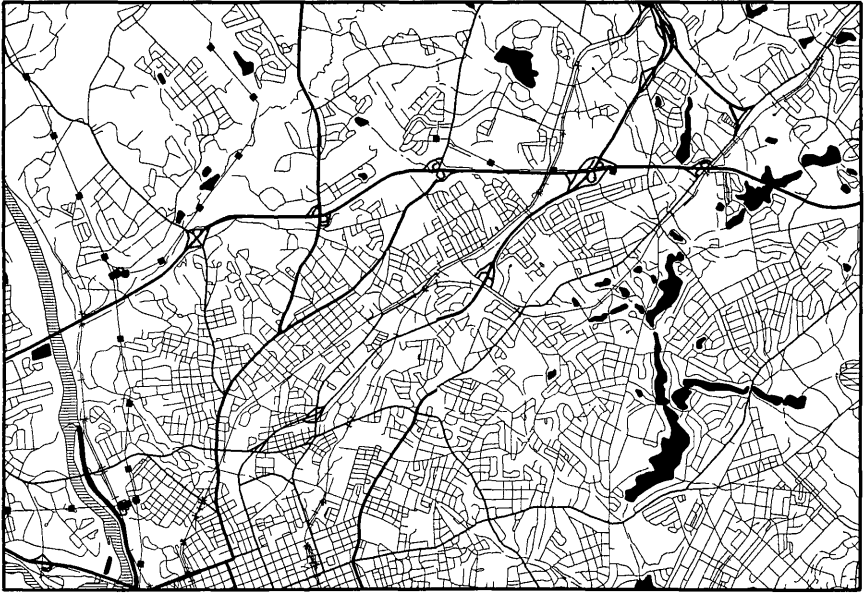


Fig. 6 1:100,000 Scale Section of Columbia, S.C.



Fig. 7 1:75,000 Scale Map of Yemassee, S.C.

is sufficiently simple that it was successfully used by members of an introductory computer mapping class last semester after a single training session. The system has also been transferred to the Eastern Mapping Division of the USGS where it will be used to generate maps for error checking and other purposes. During the next few months the programs will be ported to the UNIX environment where they will run on a network of SUN work stations. The system has proven to be particularly efficient on raster output devices and probably would be impractical for anything but simple maps of limited spatial extent on vector plotters. The success of the system clearly demonstrates that even though general purpose mapping and GIS toolboxes have evolved tremendously in their functionality there still is a place for well designed special purpose programs, especially to manage large data sets such as the DLG.

REFERENCES

Callahan, G.M. and Broome F.R. 1984, The Joint Development of a National 1:100,000 Scale Digital Cartographic Base: Technical Papers ACSM 44th Annual Meeting, pp. 246-253.

Luman, D.E. 1987, Applying USGS Digital Line Graph Data in a Microcomputer Environment: The American Cartographer, Vol. 14, pp. 321-343.

United States Geological Survey 1985, Digital Line Graphs From 1:100,000 Digital Line Graphs, National Mapping Program Technical Instructions, Data Users Guide 2, USGS, Reston, VA.

NEW YORK STATE'S DIGITAL COUNTY MAPPING PROGRAM

Ted W. Koch and William F. Johnson
New York State Department of Transportation
Albany, New York 12232

ABSTRACT

This paper details the development, design and data used for producing a statewide series of digital county maps by the New York State Department of Transportation. Three counties have been completed with others in production. Each map in the series is produced by combining digital data from the U.S. Geological Survey's 1:100,000 scale Digital Line Graphs (DLG) with data digitized from the New York State Department of Transportation's 1:24,000 7.5' quadrangles and other data compiled from recent aerial photographs and other sources. All digital information is stored in the Department's Intergraph* interactive graphics computer system and is structured for a variety of uses. Using a laser plotter, and image compositing software, converted raster files are exposed to create color-separated, press-ready negatives. The County Base Map series is part of the Department's ongoing development of a statewide digital cartographic database to be used for multi-scale map publication and Geographic Information System applications.

BACKGROUND

The New York State Department of Transportation's (NYSDOT) Mapping Services Bureau is responsible for producing and maintaining the state's several base map series. The Statewide Base Mapping Program includes 1:24,000 scale planimetric and topographic maps which are based on the USGS's 1:24,000 scale maps; 1:9600 scale city and village maps; the 1:250,000 Four Sheet State Map; and the New York State Atlas. The County Base Map (CBM) Series, which has recently gone into production, provides the needed intermediate scale component of the Statewide Base Mapping Program. The various map series in the program are interrelated, e.g. use common grids and projection, data plotted at larger scales is used at smaller scales, etc.

The NYSDOT's multi-scale base mapping program was established over twenty years ago. Unfortunately, the production of the CBM Series was deferred for many years due to other priorities and lack of funds. However, with

*Mention of firms that manufacture computer hardware and/or software, or that supply commercial mapping services or data is for descriptive purposes only and does not constitute endorsement by the New York State Department of Transportation.

renewed interest in the series from within the NYSDOT and with support and funding from the Federal Highway Administration (FHWA), design and planning work was begun in 1986, followed by the publication of Monroe County, the first map, in mid-1988. (See figure 1)

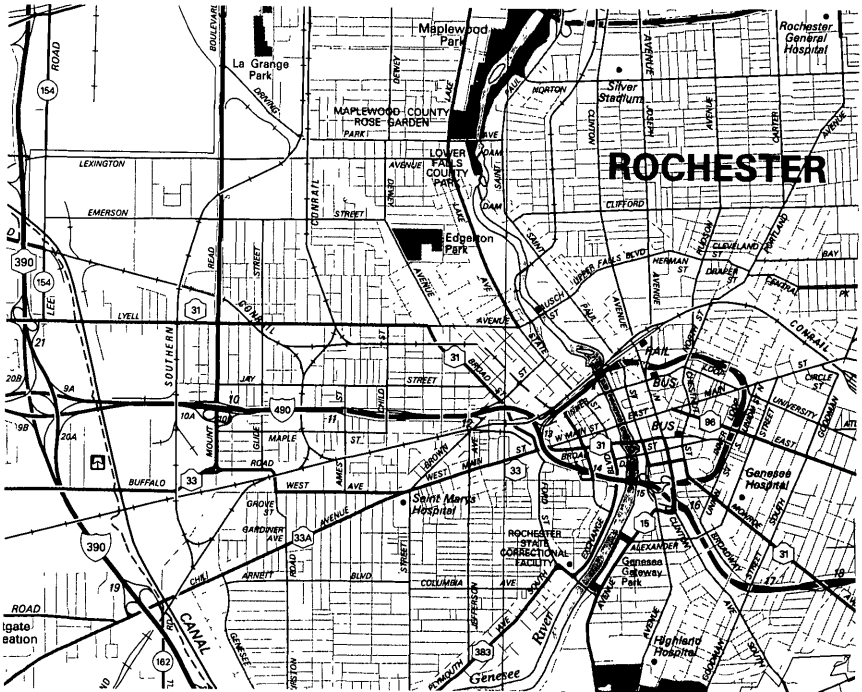


Figure 1. Black and white composite of a section of the four-color Monroe County Map at 1:75,000 scale.

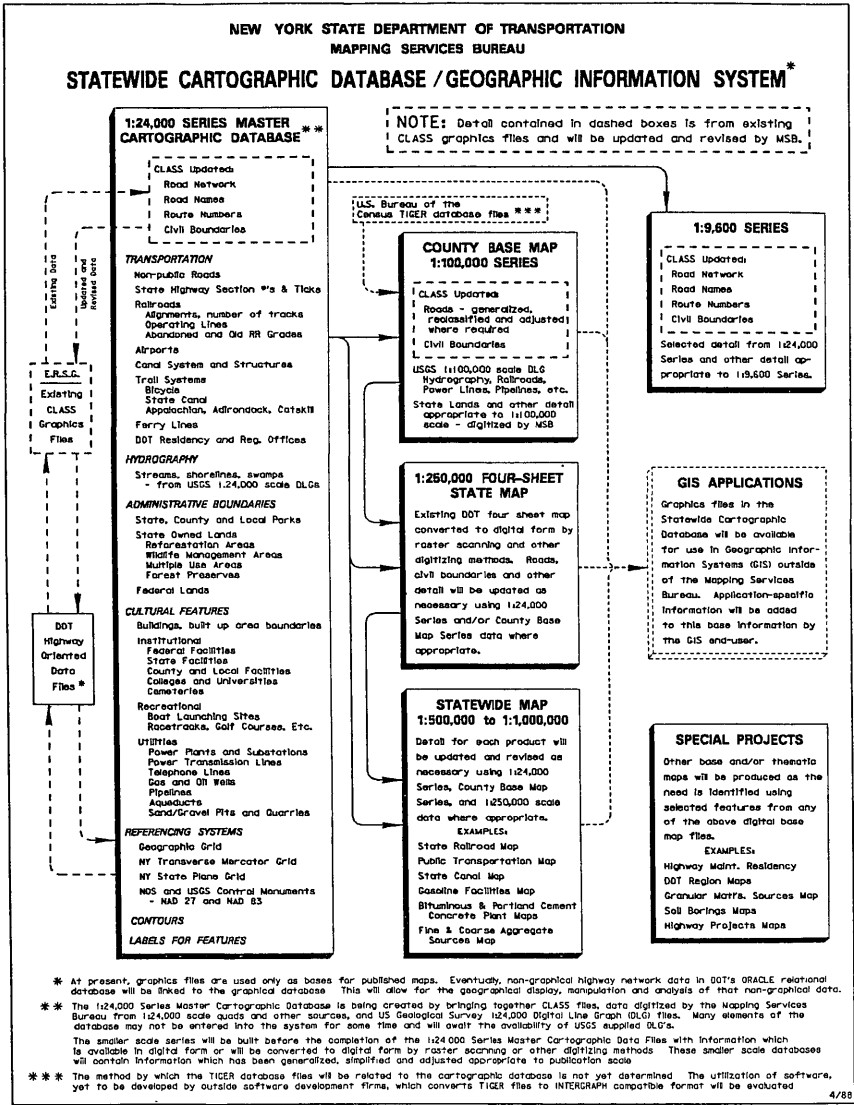
The NYSDOT's purpose in producing the long-planned CBM Series was to have a county formatted map showing in detail the components of the state's transportation network and other important base features such as civil boundaries, hydrography, and public lands. While the CBM series was designed primarily to meet the needs and requirements of NYSDOT, assuring that the maps met the needs of other New York State agencies was an important consideration.

The Mapping Services Bureau is continuing the process of converting from traditional manual production of base maps to completely automated digital production. Up to the time that digital production was started on the CBM Series in early 1987, digital methods had been concentrated on the production of large-scale photogrammetric products and on a few small scale specialty maps. Since then, we have begun to convert the 1:24,000 scale quadrangle revision process to digital methods also. Laser plotting of all revision work in this series is now routine.

DESIGN

When the CBM Series design and production goals were outlined in 1986 we committed ourselves to accomplishing the following:

- o The series would be produced digitally, from initial data input to final press-ready films for printing. With an Intergraph system available for digital production and a staff of experienced cartographers to operate the system, we felt we had the necessary resources for producing a high quality product.
- o The digital database for the CBM series had to be compatible with NYSDOT's other statewide base map series, even though these series were not yet in digital format. Compatibility would assure future use for other multi-county or regional maps. To this end, we developed a Statewide Cartographic Database / Geographic Information System plan that outlined our concept of a multi-scale digital database to support production of the Statewide Base Mapping Program. (See figure 2)
- o Utilize, to the maximum extent possible, existing digital databases to minimize the need for digitizing and other types of data entry.
- o Maps in the series would be produced at one of three scales; 1:75,000 for urban counties, 1:100,000 for moderately developed counties, and 1:125,000 for the large rural counties in the Adirondack Region of New York State. Smaller adjacent counties would be published on one sheet wherever possible.
- o Content categories would include the following: All public roads, railroads, airports, public transportation facilities, bike routes, power transmission and pipelines, civil boundaries, federal lands, state and local recreation lands, state and federal historic sites, hydrography, dams and locks, boat launch sites, and selected points of interest such as colleges, universities, hospitals and stadiums.
- o All maps would be printed in four colors, with copies available both flat and folded. To produce color-separated copy for printing, laser plotter technology would be used, if possible, to produce press-ready films. Our objective was to eliminate all labor intensive manual production techniques including; hand scribing, type stick-up, cutting open window tint negatives, and final photo-mechanical compositing of film overlays. Methods were to be developed to perform these tasks with time and money saving automated techniques.



* At present, graphics files are used only as bases for published maps. Eventually, non-graphical highway network data in DOT's ORACLE relational database will be linked to the graphical database. This will allow for the geographical display, manipulation and analysis of that non-graphical data.

** The 1:24,000 Series Master Cartographic Database is being created by bringing together CLASS files, data digitized by the Mapping Services Bureau from 1:24,000 scale quads and other sources, and US Geological Survey 1:24,000 Digital Line Graph (DLG) files. Many elements of the database may not be entered into the system for some time and will await the availability of USGS supplied DLGs.

The smaller scale series will be built before the completion of the 1:24,000 Series Master Cartographic Data Files with information which is available in digital form or will be converted to digital form by raster scanning or other digitizing methods. These smaller scale databases will contain information which has been generalized, simplified and adjusted appropriate to publication scale.

*** The method by which the TIGER database files will be related to the cartographic database is not yet determined. The utilization of software, yet to be developed by outside software development firms, which converts TIGER files to INTERGRAPH compatible format will be evaluated.

4/78

Figure 2. Statewide Cartographic Database / Geographic Information System Outline.

DIGITAL FILE CHARACTERISTICS

The CBM Series digital files are composed of six standard theme files for each county: Roads, Boundaries, Hydrography, Miscellaneous Transportation, Polygons, and Names/Design. All theme files use the same coordinate system so they register (overlay) precisely. All data is stored in Intergraph Corporation's design file format using Intergraph's Interactive Graphics Design Software (IGDS). Currently we are using IGDS version 8.8 on a

Digital Equipment Corporation (DEC) VAX 11-785.

Data in each of the six standard theme files is classified by graphic level. Additionally, several other graphic properties are used to differentiate map features, including color, line weight, line style, element class and graphic groups. Graphic definition of map features is being used as an interim approach while NYSDOT develops an agency-wide corporate attribute database.

Coordinates of all features in the files are expressed in New York Transverse Mercator (NYTM) values. NYTM is an east and west extension of Zone 18 of the Universal Transverse Mercator (UTM) projection/grid system that accommodates all of New York State in a single zone with a single origin. Coordinate values are metric.

DATA SOURCES

One of the principal activities in the planning and design of the CBM Series was evaluating and selecting existing digital databases. From a variety of databases considered, two were evaluated in detail, the 1:100,000 scale Digital Line Graph (DLG) database of the USGS, and the 1:24,000 scale Centralized Local Accident Surveillance System (CLASS) database of the NYSDOT. Ultimately, both of these data sources were chosen to form the foundation of the CBM Series digital files.

DLG Files

The DLG files are used for the Hydrography and Miscellaneous Transportation themes. DLG data has been converted to NYTM coordinates, edge matched, substantially updated, reclassified, and vertically integrated (matched) with data from other files.

CLASS Files

For the Roads and Boundaries themes, NYSDOT CLASS files were used. These files were originally table digitized from NYSDOT 1:24,000 scale quadrangles approximately 10 years ago.

For use in the CBM Series, CLASS files are updated using more recent NYSDOT quadrangles and aerial photographs, merged to form county files, reclassified to depict road jurisdiction and physical characteristics, and selectively generalized as appropriate for publication at county map scale. Civil boundaries, which are surprisingly dynamic in New York State, were updated from official sources.

All other features in the CBM files, such as state and federal lands and municipal parks, were compiled and digitized using NYSDOT 1:24,000 scale quadrangles.

All digitizing from NYSDOT 1:24,000 scale quadrangle maps was performed on high precision equipment using stable base film copies of the 7.5' quadrangles. Digitized maps were related to precise theoretical NYTM values of the

quadrangle corners for accurate digitizing control. In addition, stable base ink-on-film check plots are used to verify the accuracy of digitized alignments. All files are edge matched to form a seamless database.

Data File Evaluation

The choice of using the CLASS files over the DLG files for the Roads theme was the result of a careful comparison of the two data sets, and an estimate of the amount of work required to adapt either data set for the CBM Series. Our conclusion was that the DLG road files were suitable for the series, but would require a greater effort in sorting the roads to our classification scheme and associating them with their road names and route numbers. The CLASS files already contained these names and numbers, and we were able to programmatically convert the CLASS files to our road classification scheme based on this attribute data, a process we could not do based on attributes in the DLG road files. In addition, the CLASS files contained civil boundaries, and required no additional effort to vertically integrate roads and boundaries where they coincide.

Scale/Accuracy

The files have no expressed map scale, since all features in the files are stored in ground coordinates. However, the data sources for information in the files range from 1:24,000 to 1:100,000. Positional accuracy of features in the files is no better than the sources used for digitizing.

FEATURE TYPES

Four categories of map features are included in the CBM digital files: point, linear, area, and text/labels.

Point Features

Point features are represented at a single coordinate pair by either a cell or a symbol font. A cell consists of lines and other information which define a feature and are grouped together as a single element. A cell is stored in the file at a single point location. Cul-de-sacs at the ends of subdivision roads are represented by cells in the Roads theme file. A symbol font is a single character placed in the graphics file at a point location coordinate. The graphic representation of the character is stored in a separate font library and is displayed as the special symbol when the font library is attached to the graphics file. Symbols in open or outline form, such as route markers, use a unique font in the CBM font library, while color filled symbols, such as road rest areas, employ a different font. This allows selected symbols to be used as digital masks in a later image compositing process.

Linear Features

All linear features are represented by centerlines. Neither curve strings nor arcs have been used in the files. All linear features are stored as line strings with enough vertices to provide for reasonably smooth

bends.

Roads are stored in link/node format, with nodes (breaks) at intersections. Thus, long roads with many intersections are made of many short line strings. Some aspects of the road alignment for the published map are generalized from the original 1:24,000 CLASS files. Close alignments may be "pulled apart" so they do not touch or overlay at final published scale.

In the CBM Boundaries theme, all boundary information is kept at 1:24,000 scale accuracy specifications and is shown without generalization on the published map. Additionally, there are many instances where boundary lines coincide with other linear features, such as roads, hydrography, or other boundaries. Where boundaries coincide with another feature, such as a DLG stream alignment, we replaced the 1:100,000 scale DLG stream line with a 1:24,000 digitized line along the match zone. In this sense, we have selectively improved the DLG hydrography file to 1:24,000 scale accuracy.

In general, coincident features are stored separately with identical, but duplicate, line strings in each applicable theme file. For example, a boundary along a road is stored as a boundary in the Boundary theme file and also as a road line string in the Roads theme file. Coincident features within the same theme file are represented only once based on a hierarchy. For example, a state boundary is higher in the boundary hierarchy than a county boundary, and therefore, only the state boundary is represented along the match zone in the boundary file.

Area Features

All area feature polygons that are shown on the printed map with color fill are included in a separate Polygons theme file. The polygons are generated from line strings contained in the Hydrography and Boundaries theme files and thus duplicate the line strings from those files. Polygons are kept in a separate theme file so that the line strings can remain as simple linear elements in the original Boundaries and Hydrography theme files.

Since Intergraph polygons can only be coded for a single graphic level and do not share common edges with other polygons, cases of coincident linear features are handled through different element classes in the Polygons theme file. By this method coincident lines from different files can be built into one polygon, but can still retain the flexibility to selectively display the coincident alignments on the printed map.

Text and Labels

All text labels on the printed county map are stored in the Names/Design theme file. All text was produced digitally and was plotted with the high quality Intergraph "Bitstream" fonts. (See Figure 1) Type placement was, for the most part, performed interactively at the workstations. For populated place names Rand McNally's Randata file was used to automatically enter

names into the graphics file along with the type size codes scaled to population values. The coordinates in the Randata file define centroids of places, so final type position was adjusted interactively. Our approach has generally worked quite well for type placement. To accomplish all of this we have developed special commands to automatically set the names at the appropriate size, font, and distance from map features. Workstation operators make a final judgment on proper positioning.

We have not used the USGS Geographic Names Information System (GNIS) files to any significant extent. The GNIS file contains too little information about the size, category, and extent of features to allow for selection and classification of the names needed in the CBM series.

TOPOLOGY/ATTRIBUTES

The CBM files contain neither explicitly coded topological relationships of map features nor linkages to non-graphic attribute data. However, all files have been software checked to insure that end points of adjoining features match, and that breaks (nodes) occur properly at feature intersections. The files are clean and ready for conversion to a topologically structured database to support future applications for GIS use and analysis.

Graphic levels have been used extensively within files to permit "bulk loading" of attributes. In general, the graphics files use separate levels for each different type of feature so that all features on a level may be tagged simultaneously with attribute descriptors. We have on an experimental basis, converted the Monroe County Roads theme file to an ESRI ARC/INFO file, bypassing a SIF translation. This was accomplished by writing a Fortran program which reads the file and writes a macro to build the same graphics in ARC/INFO. Information attributes were created automatically from the data assigned to Intergraph levels. Work in this area is continuing.

FILE SIZE

Generally files for the CBM series are large. Of course, there is a wide variation in the volume of data depending on the size of the county. A single urban county may have 15 megabytes of data for all theme files combined. A range of 6-12 megabytes is expected for less urbanized counties, although we have not mapped any yet. For any county the Roads file will typically represent over half of the total county data.

LASER PLOTTING/PRINTING

For final publication all sheets in the county map series are offset printed in four colors: blue, yellow, red and dark brown. The composite film negatives used to make printing plates are created on a laser plotter which plots all information from the county files. For the Monroe County map the laser plotting was performed by

Hammond, Inc. of Maplewood, New Jersey.

Hammond uses an Intergraph system to drive an Optronics 4040 (40 inch by 40 inch) laser plotter. The plotter can operate at several different resolutions with 2000 lines per inch (12.5 microns) being the finest. Intergraph ILMS software creates the composite raster files to be output on the Optronics plotter. Through ILMS all design specifications for the printed copy, including line thickness, dashing, screening, masking, and color compositing can be accomplished. Within this environment, ILMS allows specified map features to have priority over other features. For example, route markers can be shown on a highway, with the line representing the highway broken under the marker only when plotted. In the graphics file the line remains unbroken. This masking technique is also used to show highway grade separations, and to create islands within polygon tints. Refer to Figure 1.

To achieve the highest image quality, the Monroe County map was plotted at 2000 lines per inch (12.5 microns) resolution. This yields the sharpest possible image, reducing any visible stair-stepping (aliasing) on lines and text to a minimum. For future maps we will plot at the 1000 lines per inch (25 microns) resolution for line and text, believing that this resolution will still give acceptable visual quality. In addition to the 1000 line per inch resolution, a process in the ILMS software called pixel replication can simultaneously allow plotting of polygon tints at 2000 lines per inch resolution, giving them a smoother and more even appearance than if done at 1000 lines per inch.

CONCLUSION

With the publication of the Monroe County Map in 1988 the goals we set for accomplishing the cost effective production of a fully automated and digitally stored county map have been achieved. In addition, we have built graphics files that are suitable for conversion to Geographic Information System use in the future. In fact, soon after the publication of the Monroe County Map, a Monroe County agency bought copies of all the digital files for use in developing an automated county-wide water and sewer planning and analysis system.

VECTOR-BASED COMPUTER GRAPHICS IN AUTOMATED MAP COMPILATION

C F Scheepers, Scientist
Centre for Advanced Computing and Decision Support
CSIR, P O Box 395, Pretoria 0001
Republic of South Africa

ABSTRACT

This paper describes a computer-assisted mechanism for constructing line and area symbols on maps. Vector-based computer graphics techniques such as hatching and clipping are extended for this purpose. In an effort to enhance visually the perception levels of line and area symbols, filling and following algorithms are used to place basic symbols on line and area symbols. Filling and following densities can be modified easily by changing input parameters, affording the cartographer freedom in symbol design and use.

INTRODUCTION

The application of computers and related technology to cartography has revolutionized the art and science of cartography. The digital geographic database has proved to be supreme as a storage medium for spatial information, resulting in the utilization of maps primarily for the communication of preselected spatial information. Moreover, substantial changes to the process of map compilation have also taken place. Through automation and the exploitation of computer graphics techniques, much of the tedium of manual map compilation has been relieved. Specifically, the cartographer could be assisted with the selection of cartographic features to be placed, with the design of a map symbolism, and with the creation and placement of additional features such as titles, legends, north indicators, grids, and feature name labels.

The function of maps in cartography is closely related to the general purpose of computer graphics. Whether two-dimensional or three-dimensional, or even vector or raster-based, the ultimate goal in computer graphics is to convey information graphically. Maps convey *spatial* information in a similar way. Spatial information pertaining to cartographic features of different dimensionality (point, line and area type features) is represented by means of symbols. These symbols should be designed not only to give an indication of the types of features they represent, but also to reflect characteristic properties of the features. In figure 1, examples of point, line and area symbols are illustrated. Note that *follow* and *fill* symbols are often included with line and area symbols to enhance the ability of the latter to convey information.

A SPECIFICATION FOR MAP SYMBOLISM

Before the computer-assisted mechanism for the construction of line and area symbols is presented, it is necessary to describe a method by which the map symbolism may be specified.

About line segments and pixels

The difference between vector-based and raster-based graphics is characterized by the primitive elements used for building pictures. In vector-based graphics the primitive element is the line segment, whereas the picture element (or pixel) is used in raster-based graphics.

Since point and line symbols (and the borders of area symbols) are more naturally described using a vector-based format, and since vector-to-raster conversion routines now frequently reside in hardware, the choice to describe symbols in a vector-based rather than a raster-based fashion is the least limiting. More specifically, raster technology could still be used even though symbols are described in a vector-based format.

A formal specification

The description of symbols is now formally specified in extended Backus-Naur form (Scheepers 1987b). Note that four conceptual primitives are defined to supply graphic building blocks at a somewhat higher level:

- *dots*: line segments with zero length;
- *segments*: visible or invisible line segments;
- *arcs*: a limited number of line segments approximating an arc;
- *text*: characters compiled from sets of line segments.

Key :

```

<...>   Concept
[...]   Optional
{...}+  Set of 1 or more
::=     Consists of / is defined as
|       Choice
terminals Without brackets

```

```

<point symbol> ::= <std_symbol> <placement>
<placement>   ::= centered | standing

```

```

<line symbol> ::= <line> [<follow symbol>] | [<line>] <follow symbol>
<line>       ::= single <attribute> | double <attribute>
<follow symbol> ::= <std_symbol> <follow type> <orientation> <overlap>
<follow type> ::= true | visual
<orientation> ::= perpendicular | upright | parallel
<overlap>     ::= permitted | forbidden

```

```

<area symbol> ::= <area> [<border>] | [<area>] <border>
<area>       ::= <fill symbol> | <hatched>
<fill symbol> ::= <std_symbol> <clipping>
<hatched>    ::= hatch | crosshatch
<clipping>   ::= clip | don't clip
<border>     ::= <line symbol>

```

```

<std_symbol> ::= {<picture element>}+
<picture element> ::= <primitive> <attribute>
<primitive>    ::= dot | segment | arc | text
<attribute>   ::= colour linetype linewidth textsize

```

Terminals are defined graphically in figure 2.

Parameters for the specification

Additional information would be needed during the actual construction of symbols. This information may be imported using the following parameters (see figure 3):

- *point symbols*:
 - The orientation and relative size of point symbols.

- *line symbols*:
 - The distance between double lines.
 - The distance between individual follow symbols.
 - The size of follow symbols relative to basic symbols.
- *area symbols*:
 - The orientation of hatch lines and the distance between them.
 - The spacing parameters for regular fill patterns.
 - The orientation and relative size of fill symbols.

THE CONSTRUCTION OF LINE SYMBOLS

From the specification in the previous section, a line symbol may be defined in two ways. Firstly, available attributes may be used to construct single or double lines and, secondly, a set of follow symbols may be used to construct the line symbol. Note that usage of the one method does not preclude the use of the other.

If the second method is used, the follow type (*true* or *visual*) and the orientation of follow symbols (*perpendicular*, *upright* or *parallel*) should be considered. Furthermore, if both methods are used simultaneously, the overlap specification (*permitted* or *forbidden*) should also be catered for.

Single and double lines

The construction of single lines is trivial. Double lines, on the other hand, require careful consideration. Apart from using the available attributes to distinguish between different double line symbols, a distance parameter should also be incorporated. Suppose that the cartographic line feature to be represented is described by n line segments with coordinates $(x_0, y_0) \dots (x_n, y_n)$ as illustrated in figure 4. To determine the coordinates $(x'_0, y'_0) \dots (x'_n, y'_n)$ representing one line of the double line symbol, two cases are considered:

- Determining the start point (x'_0, y'_0) and end point (x'_n, y'_n) .
- Determining (x'_i, y'_i) for $i = 1, 2, \dots, n - 1$, the rest of the coordinates.

To determine the coordinates representing the second line of the double line symbol, a similar approach is taken.

The start and end points. The coordinate of the start point (x'_0, y'_0) in figure 4 is given by the following formulas:

$$\begin{array}{rcl}
 x'_0 & = & d \cos \alpha \\
 & = & d \cos(\frac{\pi}{2} - \theta_1) \\
 & = & d \sin \theta_1 \\
 & = & d(y_1 - y_0)/s
 \end{array}
 \quad \text{and} \quad
 \begin{array}{rcl}
 y'_0 & = & b \sin \alpha \\
 & = & d \sin(\frac{\pi}{2} - \theta_1) \\
 & = & d \cos \theta_1 \\
 & = & d(x_1 - x_0)/s
 \end{array}$$

where d is half the perpendicular distance between the double lines and

$$s = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}.$$

The end point (x'_n, y'_n) is determined in a similar fashion by replacing (x_0, y_0) and (x_1, y_1) with (x_{n-1}, y_{n-1}) and (x_n, y_n) .

The other coordinates. To simplify the discussion which follows, assume that the coordinate (x_i, y_i) is placed at the origin of a Cartesian coordinate system. Then, the coordinate of (x'_i, y'_i) is easily determined by intersecting the two lines A ($y \cos \alpha_1 - x \sin \alpha_1 - d = 0$) and B ($y \cos \alpha_2 - x \sin \alpha_2 - d = 0$) in figure 4.

Let the distances from the origin to (x_{i+1}, y_{i+1}) and (x_{i-1}, y_{i-1}) be $s_1 = \sqrt{x_{i+1}^2 + y_{i+1}^2}$ and $s_2 = \sqrt{x_{i-1}^2 + y_{i-1}^2}$ respectively, then since $\cos \alpha_1 = x_{i+1}/s_1$, $\cos \alpha_2 = x_{i-1}/s_2$,

$\sin \alpha_1 = y_{i+1}/s_1$, and $\sin \alpha_2 = y_{i-1}/s_2$, the equations above may be rewritten as

$$\begin{aligned}yx_{i+1} - xy_{i+1} - ds_1 &= 0, \\ \implies y &= \frac{xy_{i+1} + ds_1}{x_{i+1}}\end{aligned}\tag{1}$$

and

$$\begin{aligned}yx_{i-1} - xy_{i-1} - ds_2 &= 0, \\ \implies y &= \frac{xy_{i-1} + ds_2}{x_{i-1}}.\end{aligned}\tag{2}$$

Using (1) and (2), the intersection point (x'_i, y'_i) is determined as follows:

$$\begin{aligned}\frac{xy_{i+1} + ds_1}{x_{i+1}} &= \frac{xy_{i-1} + ds_2}{x_{i-1}}, \\ \implies (xy_{i+1} + ds_1)x_{i-1} &= (xy_{i-1} + ds_2)x_{i+1} \\ \implies xx_{i-1}y_{i+1} + dx_{i-1}s_1 &= xx_{i+1}y_{i-1} + dx_{i+1}s_2 \\ \implies x &= \frac{d(x_{i+1}s_2 - x_{i-1}s_1)}{x_{i-1}y_{i+1} - x_{i+1}y_{i-1}},\end{aligned}$$

provided that $x_{i-1}y_{i+1} - x_{i+1}y_{i-1} \neq 0$. Similarly, $y = d(y_{i+1}s_2 - y_{i-1}s_1)/(x_{i-1}y_{i+1} - x_{i+1}y_{i-1})$ provided that $x_{i-1}y_{i+1} - x_{i+1}y_{i-1} \neq 0$.

If $x_{i-1}y_{i+1} - x_{i+1}y_{i-1}$ is indeed equal to zero, then the lines A and B are collinear, and the intersection is not calculated.

Line following

Follow symbols are placed on line symbols with a predetermined distance between consecutive symbols. This distance may be interpreted in two ways. With a *true* distance measurement, the distance between consecutive symbols is measured along the curvature of the line. On the other hand, if the distance is representative of the radius of a circle placed on one follow symbol, with the next follow symbol being placed at the intersection of this circle with the curvature of the line, the distance is called a *visual* distance measurement.

True follow distance. The placing of follow symbols according to the true distance measurement presents few problems. Line segment lengths are merely accumulated until a length greater than the true follow distance is found. The follow symbol should then be placed on the last line segment used during the accumulation process. Note that if a single line segment is longer than the required distance, more than one follow symbol might have to be placed on that particular segment.

Visual follow distance. With the placing of visual type follow symbols the distances between a previously placed follow symbol and the endpoints of consecutive line segments are compared to the required visual follow distance. If any of these distances exceed the visual follow distance, a new follow symbol should be placed. Once again, the symbol is placed on the last line segment under consideration, or more specifically, at the intersection of the circle and the line segment as illustrated in figure 5. A formula for determining (x, y) may be derived as follows (Scheepers 1987b):

Line equation:

$$\begin{aligned}y &= mx + c \\ &= x \tan \theta + c && \text{if } \cos \theta \neq 0 \\ &= x \frac{\sin \theta}{\cos \theta} + c \\ &= \frac{x \sin \theta + z}{\cos \theta},\end{aligned}\tag{3}$$

with

$$\begin{aligned}
 z &= c \cos \theta \\
 &= (y - x \tan \theta) \cos \theta \\
 &= y \cos \theta - x \sin \theta.
 \end{aligned} \tag{4}$$

Circle equation:

$$y^2 = r^2 - x^2. \tag{5}$$

(3)² - (4):

$$\begin{aligned}
 0 &= \left(x \frac{\sin \theta}{\cos \theta} + \frac{z}{\cos \theta}\right)^2 - (r^2 - x^2) \\
 &= x^2 \frac{\sin^2 \theta}{\cos^2 \theta} + x \frac{2z \sin \theta}{\cos^2 \theta} + \frac{z^2}{\cos^2 \theta} - r^2 + x^2 \\
 &= x^2 \sin^2 \theta + x^2 \cos^2 \theta + 2xz \sin \theta + z^2 - r^2 \cos^2 \theta \\
 &= x^2 + 2z \sin \theta x + z^2 - r^2 \cos^2 \theta.
 \end{aligned}$$

Thus:

$$\begin{aligned}
 x &= \frac{-2z \sin \theta \pm \sqrt{4z^2 \sin^2 \theta - 4(z^2 - r^2 \cos^2 \theta)}}{2} \\
 &= -z \sin \theta \pm \sqrt{z^2 \sin^2 \theta - z^2 + r^2 \cos^2 \theta} \\
 &= -z \sin \theta \pm \sqrt{z^2 (\sin^2 \theta - 1) + r^2 \cos^2 \theta} \\
 &= -z \sin \theta \pm \sqrt{-z^2 \cos^2 \theta + r^2 \cos^2 \theta} \\
 &= -z \sin \theta \pm \cos \theta \sqrt{r^2 - z^2}.
 \end{aligned} \tag{6}$$

(6) into (3) gives:

$$\begin{aligned}
 y &= \frac{(-z \sin \theta \pm \cos \theta \sqrt{r^2 - z^2}) \sin \theta + z}{\cos \theta} \\
 &= \frac{-z \sin^2 \theta}{\cos \theta} \pm \sin \theta \sqrt{r^2 - z^2} + \frac{z}{\cos \theta}.
 \end{aligned}$$

Let $s = \sqrt{r^2 - z^2}$, then, from (4):

$$\begin{aligned}
 y &= \frac{-(y_1 \cos \theta - x_1 \sin \theta) \sin^2 \theta}{\cos \theta} \pm s \sin \theta + \frac{y_1 \cos \theta - x_1 \sin \theta}{\cos \theta} \\
 &= x_1 \frac{\sin^3 \theta}{\cos \theta} - y_1 \sin^2 \theta \pm s \sin \theta + y_1 - x_1 \frac{\sin \theta}{\cos \theta} \\
 &= x_1 \frac{\sin \theta}{\cos \theta} (\sin^2 \theta - 1) - y_1 (\sin^2 \theta - 1) \pm s \sin \theta \\
 &= \frac{-x_1 \sin \theta \cos^2 \theta}{\cos \theta} + y_1 \cos^2 \theta \pm s \sin \theta \\
 &= y_1 \cos^2 \theta - x_1 \sin \theta \cos \theta \pm s \sin \theta.
 \end{aligned}$$

Continuing with (6):

$$\begin{aligned}
 x &= -z \sin \theta \pm s \cos \theta \\
 &= -(y_1 \cos \theta - x_1 \sin \theta) \sin \theta \pm s \cos \theta \\
 &= x_1 \sin^2 \theta - y_1 \sin \theta \cos \theta \pm s \cos \theta,
 \end{aligned}$$

where

$$\begin{aligned} s &= \sqrt{r^2 - z^2} \\ &= \sqrt{r^2 - (y_1 \cos \theta - x_1 \sin \theta)^2} \\ &= \sqrt{r^2 - (y_1^2 \cos^2 \theta - 2x_1 y_1 \sin \theta \cos \theta + x_1^2 \sin^2 \theta)}. \end{aligned}$$

The orientation of follow symbols

The orientation of a particular follow symbol is easily determined. If the required orientation is *upright*, the symbol is merely placed in position. If, on the other hand, the orientation is either *parallel* or *perpendicular*, the orientation of the line segment on which the symbol is to be placed determines the orientation of the follow symbol.

Overlap

The overlap specification is used only if both methods for defining a line symbol is used simultaneously. If overlap is *forbidden*, the single or double line needs to be interrupted in the vicinity of each follow symbol such that no overlap occurs (indeed a form of clipping). One possible solution to this problem is to determine a bounding circle around each follow symbol, and to intersect this circle with the appropriate line segments of the single or double line. For this purpose an extension of the formula for placing visual type follow symbols may be used.

THE CONSTRUCTION OF AREA SYMBOLS

The construction of vector-based area symbols presents some very interesting problems. The areal extent of these symbols requires the use of hatching (or filling) algorithms. Furthermore, matters are complicated by the fact that cartographic regions might include 'holes' or *islands*.

From the specification for map symbolism presented earlier, an area symbol may be defined in two ways. Firstly, the borders of the area symbol could be constructed as if these borders were line symbols, and secondly, the inside of the closed region could be filled with hatch lines or with fill symbols. The type of hatch pattern (*hatch* or *crosshatch*) should be considered. Note that using the one method does not preclude the use of the other.

If both methods are used simultaneously clipping (*clip* or *don't clip*) will also have to be considered.

Hatching and filling

Traditionally, multiple simply-closed polygons with non-intersecting edges are used to approximate geographical regions by a single primitive. The lists of vertices representing the polygons are generally ordered in such a way that the *inside* of the region is implicitly defined (for example, outer boundary clockwise, all inner boundaries counter-clockwise). The methodology propagated here is to subdivide or partition regions into more manageable areas prior to hatching or filling.

The PMP partitioning algorithm. If a polygon P is considered topologically, three types of vertices can be identified with respect to the y -axis. A vertex V_i of P is called a *peak* if both $y(V_{i-1})$ and $y(V_{i+1})$ are less than $y(V_i)$, and it is called a *pit* if both $y(V_{i-1})$ and $y(V_{i+1})$ are greater than $y(V_i)$ (Cromley 1984). Vertices that are neither peaks nor pits are referred to as *regular* vertices. A polygon containing only one peak and one pit is called *monotone* (Lee 1981).

Following this terminology, define a *peak of type-1* to be a peak of the inside and a *peak of type-2* to be a peak of the outside of the area of interest as indicated in

figure 6. Similarly, let a pit of the inner region be called a *type-1 pit* and a pit of the outer region be a *type-2 pit*. Furthermore, define a *pseudo-monotone polygon (pmp)* to be a polygon that contains exactly two non-crossing, non-descending routes from its minimum to its maximum vertex.

If the region of interest in figure 6 is now considered to be a piece of paper, possibly with holes cut into it, then using a pair of scissors, it would be simple to 'cut away' *pmp*'s by cutting from every type-2 peak or pit in a horizontal direction until the edge of the paper in that direction is reached. The result of this cutting would yield six pieces of paper without any holes in them, all *pmp*'s, or in the general case:

$$n(\text{pmp}) = n(\text{type-2 peaks} + \text{type-2 pits}) + 1 - n(\text{islands})$$

Note that the actual cutting direction is arbitrary as a change of direction would result in the same number of *pmp*'s. Also note that the restriction placed by identifying peaks and pits *with respect to the y-axis* and cutting in a *horizontal direction* is also arbitrary, since the polygons representing a region to be partitioned can always be rotated if required.

Filling with hatch lines. Consider figure 7, where a hypothetical left hatch limit is illustrated. Instead of intersecting each hatch line with this limit to determine the hatch line end points, an incremental displacement along each edge is calculated (see also Brassel 1979, Cromley 1984 and Scheepers 1987a).

Let L_i denote the left end points of hatch lines. Then, by calculating $x(L_1)$ and $y(L_1)$ once for each hatch limit, and by repeatedly adding d_x and d_y , it is fast and simple to determine the other values of L_i . Assume that apart from being parallel to the x-axis of a Cartesian coordinate system, the hatch lines also have integer y-axis values.¹ Let \lfloor and \lceil represent floor and ceiling functions, then from this assumption it follows that $y(L_1) = \lceil(y(A))$. Let $DX = x(B) - x(A)$ and $DY = y(B) - y(A)$. Then

$$\frac{x(L_1) - x(A)}{y(L_1) - y(A)} = \frac{DX}{DY}$$

and therefore

$$x(L_1) = x(A) + (y(L_1) - y(A)) \frac{DX}{DY}.$$

Similarly, $x(L_2) = x(A) + (y(L_2) - y(A)) \frac{DX}{DY}$ and from figure 8 and the assumption, it follows that $d_y = y(L_2) - y(L_1) = 1$ and $d_x = x(L_2) - x(L_1) = \frac{DX}{DY}$.

Hence, the left end points of hatch lines are:

$$x(L_i) = x(L_{i-1}) + d_x$$

$$y(L_i) = y(L_{i-1}) + 1$$

for all $i = 2 \dots \mu$, $\mu = \lfloor(y(B)) - \lceil(y(A)) + 1$.

The same method is used to determine the coordinates of the right end points of hatch lines.

Filling with symbols. A similar approach to the one discussed above is taken to construct a fill pattern. First, *conceptual hatch lines* are determined. These lines are used to position fill symbols incrementally to form a regular pattern.

¹This assumption can be made without a loss of generality, since the vertices of the polygons representing the region can always be scaled accordingly.

Let s represent the spacing between consecutive fill symbols measured along conceptual hatch lines, and define an indentation factor i/o in terms of s , where i represents the number of intervals in s , and o the *indentation* (in terms of i) between consecutive hatch lines (see figure 7).

If P_j are the points on a conceptual hatch line where fill symbols should be placed, the following procedure may be used to calculate the coordinates of P_j . (Assume that P_j have integer x -axis values, since if this was not the case, scaling by s could easily be executed. Then by determining the coordinates of P_1 once for each conceptual hatch line, and repeatedly adding $d_x = x(P_2) - x(P_1)$, it is simple and fast to determine the other values of P_j .)

Procedure:

- Calculate $\beta = y(A) \bmod i$ and $\delta = \lceil (x(A) + \frac{\beta}{i}) \rceil$.
- Calculate the *indentation function* $\Psi(i, o)$:

$$\Psi(i, o) = \begin{cases} \frac{\beta}{i} & \text{for } \lceil (x(A)) \rceil = \delta \\ -\frac{(\beta \times o) \bmod i}{i} & \text{otherwise.} \end{cases}$$

- Let $DX = x(B) - x(A)$ and determine μ , where:

$$\mu = \frac{\lceil (x(A)) \rceil + \Psi(i, o) - x(A)}{DX}$$

- Calculate ν , the relationship between one spacing increment and distance AB :

$$\nu = \frac{x(P_2) - x(P_1)}{DX}$$

- Use μ and ν to determine the first point:

$$x(P_1) = x(A) + \mu DX$$

$$y(P_1) = y(A) + \mu DY$$

where $DX = x(B) - x(A)$ and $DY = y(B) - y(A)$.

- Determine the rest of the points:

$$x(P_j) = x(P_{j-1}) + \nu DX$$

$$y(P_j) = y(P_{j-1}) + \nu DY$$

for all $j = 2 \dots \gamma$, where $\gamma = \lceil (x(B) + \frac{\beta}{i}) \rceil - \lceil (x(A) + \frac{\beta}{i}) \rceil$.

Clipping fill symbols

The clipping specification is used only if both border and inner area symbolization are used in the definition of an area symbol. If clipping is required (*clip*), the approach taken depends on whether follow symbols are used in the border symbolization. If follow symbols are not used, an algorithm for polygonal clipping of polygons should be used (Matthew 1985). If, on the other hand, follow symbols are used along the borders, no clear-cut solution exists. One possibility would be to construct a conceptual double line symbol on the border of the area symbol, and to use the inner line of this double line to clip against.

CONCLUDING REMARKS

This paper has presented a computer-assisted mechanism for constructing vector-based line and area symbols on maps. A specification for map symbolism has also been described. This specification was used as a reference basis for the aforementioned presentation.

The author wishes to thank Werner Strydom for preparing the illustrations.

REFERENCES

- BRASSEL K E, FEGEAS R (1979). "An algorithm for shading of regions on vector display devices", *Computer Graphics*, **13** : 2, 126-133.
- CROMLEY R G (1984). "The peak-pit-pass polygon line-shading procedure", *The American Cartographer*, **11** : 1, 70-79.
- LEE D T (1981). "Shading of regions on vector display devices", *Computer Graphics*, **15** : 3, 37-44.
- MATTHEW A J (1985). "Polygonal clipping of polylines", *Computer Graphics Forum*, **4**, 407-414.
- SCHEEPERS C F (1987A). "Polygon shading on vector type devices", *Quaestiones Informaticae*, **5** : 2, 46-55.
- SCHEEPERS C F (1987B). "n Vektorbenadering tot grafiese voorstelling in rekenaar-ondersteunde kartografie", *CSIR CACDS Technical Report TWISK 571*, Pretoria, November 1987, 168pp, (MSc dissertation in Afrikaans).
- SCHEEPERS C F (1988). "Computer-assisted map symbolism", *Proceedings: 1988 ACSM-ASPRS Annual Convention*, **2** : Cartography, 47-56.

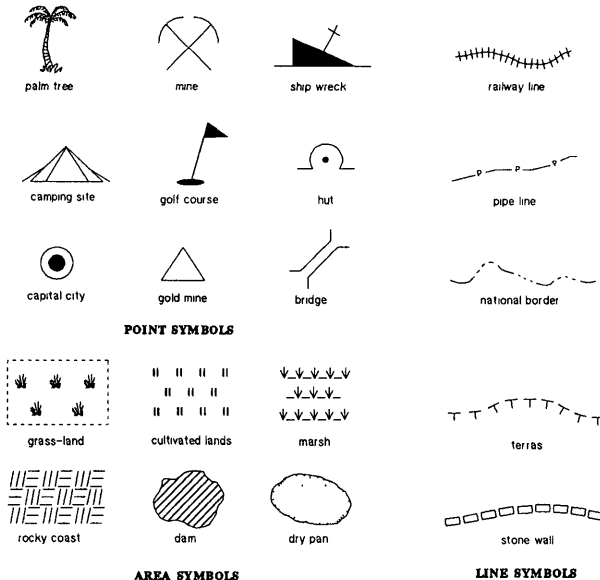


Figure 1: Examples of symbols

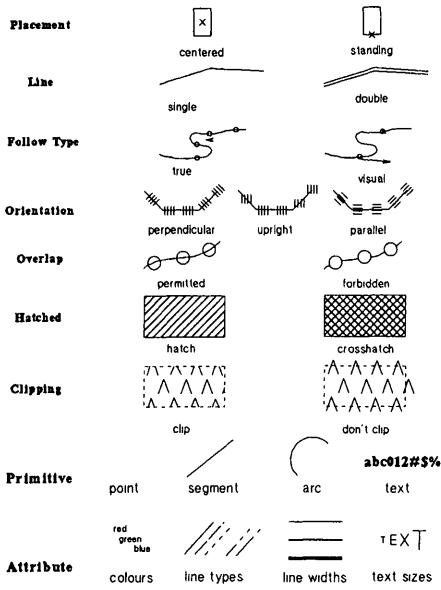


Figure 2: Terminals

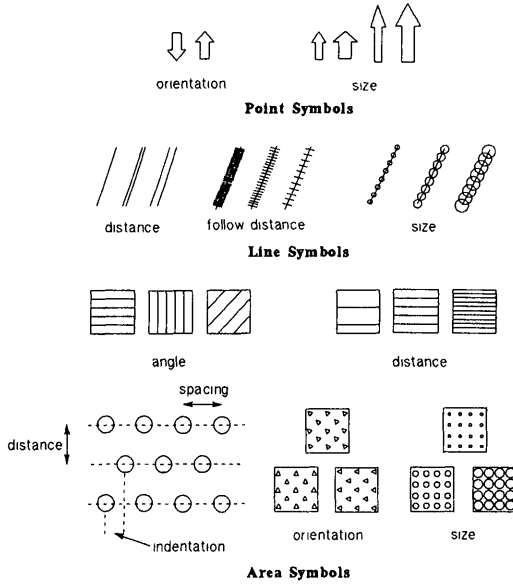


Figure 3: Parameters

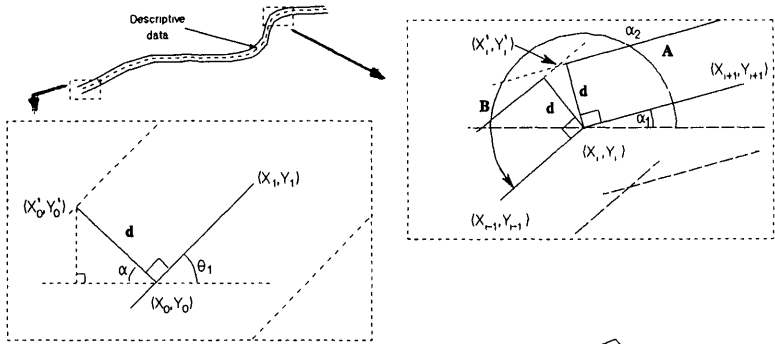


Figure 4: Double lines

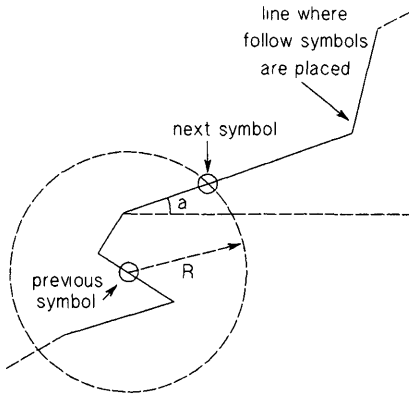


Figure 5: Visual follow symbol placement

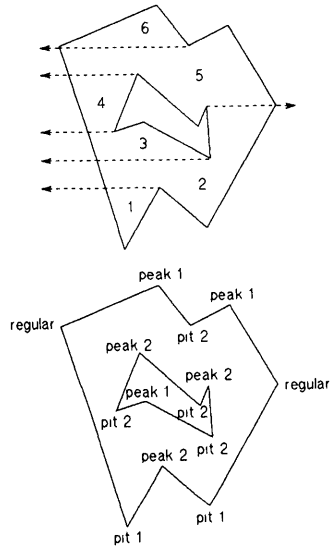
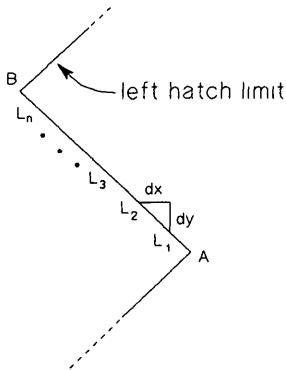
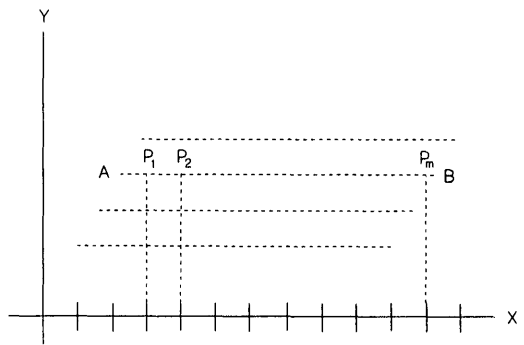


Figure 6: Polygon partitioning



Hatching



Filling

Figure 7: Hatching and filling

First UNIX, then UGIX

D.W. Rhind, J.F. Raper and N.P.A. Green

Department of Geography, Birkbeck College,
University of London,
7-15 Gresse Street, London W1P 1PA, UK

ABSTRACT

This paper proclaims the need for increased standardisation within the field of GIS and defines some assumptions about the way in which GIS systems will evolve and develop if insufficient attention is paid to standardisation. Building upon these assumptions, we describe how the existing GIS Tutor can be expanded into forming a vital part of a Universal Geographic Information eXecutive (or UGIX). The initial role of this is to act as a 'friendly front end' to any existing GIS and to free the user from having to learn new conventions, rules and grammar every time s/he works on a different system. Subsequent, higher order aims are also described.

STANDARDISATION, USERS AND VENDORS

We begin with a conjecture : the simpler and more standard are computer systems, the more readily they are used and the more choice of system is open to the user. Standardisation, at least of basic tools and methods of using them, is therefore 'a good thing', so far as the user is concerned.

A major theme of the last 10 years in the data processing industry as a whole has been that of standardisation; Nash and Redwine (1987) have identified over 1000 software-related standards in the USA alone. Perhaps the most striking example thus far has been the popularisation and gradual acceptance of the UNIX operating system and, latterly, the requirement in many defence contracts that the systems are POSIX - compliant. In the medium to longer term, Open Systems Interconnect (OSI) might be even more important. Though some GIS now capitalise upon UNIX, some provide querying through the Structured Query Language (SQL) and some systems use the Graphics Kernel System (GKS) or the Programmers' Hierarchical Interactive Graphics System (PHIGS) as graphic output mechanisms, no significant standardisation has gone on within the GIS field except in terms of data description and transfer protocols (e.g. ACA 1988 and OS 1987).

It is worth considering why this should be so and, in so doing, noting the present dominance of commercial concerns in the field. This is a recent phenomenon : the primary difference between Rhind's 1981 and 1987 reviews of the GIS field in the UK, for instance, was the almost total lack of any commercial presence observable at the earlier date yet the domination of the field by such interests at the latter. What we have seen, therefore, is an emerging market place characterised by increasingly strong competition amongst an increasing number of vendors. No one of these vendors presently seems prepared (or, in some cases, is financially

able) to move to agreed standards - even if the latter existed.

In reality, we should not be surprised at a lack of standardisation : the user base is manifestly so diverse, rapidly expanding and disorganised that it has been difficult to define a set of procedures, addressing mechanisms and conventions which cover user needs, let alone exert pressure on the vendors (except through industry - wide groups, of which the National Joint Utilities Group in the UK is a striking example). As a consequence, the vendors have little incentive to offer standardisation. Indeed, taking a longer term view, the users can be considered to be enjoying a period of warfare amongst vendors which will, through the operation of the market, result in a few successful suppliers; the products from the survivors could well form the de facto standards in years to come. Adopting this viewpoint, we - as users - should simply sit back and wait for Adam Smith's "invisible hand" to ensure the appearance of a set of standards.

We eschew this passive approach : we hold that the user has a right, even an obligation, to assist and encourage vendors in moving towards standardisation - provided that this does not stultify new developments. Without such efforts, a small number of vendors may come to dominate the market with different proprietary solutions and to 'lock in' the user to their particular products. Precisely this occurred in the 1960s and 1970s in data processing as a whole. We therefore advocate an activist and even interventionist approach, stressing the primacy of the user rather than the supplier. We make several assumptions which seem reasonable and proceed from these to recommending a course of action.

The first of our assumptions is that many GIS users will increasingly regard their data bases as a long term and possibly appreciating asset; in contrast, they will treat their GIS software as an asset which is depreciated normally. At present, this distinction is difficult since data are often intimately wrapped up in proprietary features in any one GIS. The second of our assumptions is that competitive pressures will force convergence between the solutions being offered by vendors, at least at the levels of data structure and functionality; Rhind and Green (1988) have summarised the advantages of different data structures and GISWorld (1988) has published a table of functionality claimed by vendors which seems to demonstrate the progression to functional equivalence as being underway.

Given all of the above, we anticipate that users - in GIS as elsewhere - will seek freedom to purchase the best deal available as they purchase second and third generation GISs (perhaps to meet new tasks or as software suppliers go out of business) and as costs and capabilities change. At present, this is rendered impossible by the different functionality, different data structures and the often idiosyncratic and painfully acquired knowledge of how to 'drive' any one system. To facilitate the change requires at least (as we argue later) an 'intelligent front end' which can speak to all GISs in their own command languages yet which can be instructed by the user in a universally accepted 'geospeak'.

Before we set out the components of a GIS which are amenable to standardisation, however, we describe how we see the market developing; this has important implications for the starting point and capabilities of the UGIX which we describe later in the paper. To do this, we extrapolate from the example of a more mature and widely used product than any present-day GIS.

dBASE AS AN EXAMPLAR FOR THE GIS MARKET

The most dramatic change in computing in the last 10 years has been the growth of local computing power in comparison with that in centralised machines. Thus the availability of cheap yet powerful micros fueled and, in turn, benefited from the availability of general purpose software packages. Some indications that the GIS market is already going much 'more personal' already exist : the dramatic success of ESRI, for instance, in selling 1300 copies of PC ARC/INFO in its first year, compared to about 500 copies of mini- and mainframe versions over four years, when allied to the success of SPANS and other micro- based and workstation- resident systems, suggests that GIS is merely following the trend set by systems such as Wordstar, Lotus 1-2-3, dBase and many others.

Though the analogy of its evolution with that of GIS systems is not exact, the story of dBASE is directly relevant to our concerns, not least because it has now sold over 100,000 copies and has become nearly ubiquitous. It was 'invented' by Wayne Ratcliff, being developed in his spare time to keep track of football results for the office sweepstake system. The result was entitled Vulcan and, when marketed, sold very badly. Only when Ashton-Tate took over the marketing did matters change; they initially sold it as dBASE II for use on CP/M machines. Later, Ashton-Tate bought the rights to dBASE though Ratcliff stayed on as Vice-President and in charge of development for version III. Following disputes about the way in which dBASE should develop, Ratcliff left the company. dBASE IV has recently been launched.

The nature of the product has evolved dramatically over the years. Version II was essentially a programmer's toolbox. It consisted of just over 100 commands, each of which was activated via use of the dot prompt. Restrictions were numerous e.g. a limit of 65,535 records, each of which could have up to 32 fields; a very limited form of relating together two files was provided. Version III, in contrast, provided a menu- driven package called the Assistant; it expanded the number of commands by about 35%, introduced set relational capabilities and relaxed many of the more irksome restrictions and facilitated the creation of user-designed screens. The extension, dBASE III Plus, introduced more powerful commands for programmers and improved the Assistant. Finally (at least thus far), dBASE IV provides multiple user interfaces, an increase in speed and improved networking capabilities. In particular, the Assistant has been replaced by the Control Centre which enables those users who so choose to use dBASE as an entirely menu-driven system. It provides Query By Example capabilities and an applications generator.

An SQL (Structured Query Language) interface has been added whilst the original dBASE idiosyncratic commands may still be used.

Several relevant conclusions may be drawn from the dBASE story. These include :

- that early systems which become successful encourage both lower cost clones and 'add-ons'; through this, they may become de-facto standards
- though many successful systems start out designed for experts, they end up catering for a mass market
- the size of evolving packages normally gets larger and larger as the price both of ensuring upward compatibility and of providing new features. An important (and expensive) part of the additional features is likely to be an accomodation of standards initially ignored or recently promulgated.
- fortunately, the recent annual growth in computing power per unit cost has exceeded the rate of growth of size of software systems and also the size of 'average' applications (note that the latter statement does not apply to the largest GIS applications, leading to a divergence in the computing needs of 'average' and large scale GIS users (e.g. global modellers)). dBASE succeeded by enabling average size applications to be run - and continue to be run as these applications became larger - on contemporary micro- computers.
- in dealing with a mass market product, superb documentation, highly robust software, training materials and a secondary 'value added' industry are essential to provide success
- in many respects, the adoption and spread of data base systems developed along similar, but earlier, lines to those of GISs. Given this precedent and the ready availability of mass market graphical, data base and other software, it seems reasonable to expect PC GISs soon to be selling for little more than the combined cost of dBASE and, say, the Harvard Graphics package (c. #750 at list price in the UK). If standard data bases are also available, the consequences of such pricing for GIS sales will be immense.

GIS ELEMENTS AMENABLE TO STANDARDISATION

These would seem to be :

- (i) the terminology used to describe individual functions, data elements, etc
- (ii) the form and structure of data dictionaries
- (iii) a set of standard tasks for testing GIS, with mathematically provable end - results

- (iv) a library of proved routines, using the best available algorithms. This would be the equivalent of the Numerical Algorithms Group (or NAG) Library. An early attempt to bring this about was the Geographical Algorithms Library or GAG (Campbell 1977)
- (v) standard human interfaces to GIS; as Rhind and Green (1986) and others have pointed out and as dBASE, Domesday (Rhind et al 1988) and other systems have implemented, multiple access routeways are needed for different users and types of applications. These will inevitably include WIMPs (with standard ikons), an SQL interface and simple sets of menus. More important, it should include a 'Geospeak' language since SQL is ill-suited for spatial queries yet many individuals prefer to express their requests in a written rather than a graphic language. Egonhofer and Frank (1988), Palmer and Frank (1988) and Goh (forthcoming) have begun the process of designing spatial query languages.
- (vi) data transfer formats and protocols. Much work has already been carried out in this area.

From all of this (and other evidence), we conclude that mass market GISs are likely to form THE major growth area and that this will be based upon Mac, PS/2 or DEC architectures. This is manifestly a technical possibility : existing packages with the power of 386 processors, allied to Winchester - type storage of 100 to 300 Mb and archival storage of 600Mb per exchangeable CD-ROM, give capabilities available a few years ago to only the most privileged. We anticipate, therefore, a growing number of PC and workstation - based GIS over the next few years. Thus we believe that action is necessary now and that the best way to bring about standardisation should be :

- (i) international action, involving multi-disciplinary inputs, to tackle items (i) to (iv) and (vi) in the list of elements amenable to standardisation.
- (ii) a concentration on 'binding in' UGIX to the micro-based systems (although it must also be available as a micro-based 'front end' which can converse with and issue instructions to mini- and mainframe-based GIS). This therefore addresses the final element in our list and the rest of this paper is devoted to this topic.

THE uNIVERSAL gEOGRAPHIC iNFORMATION eXECUTIVE (OR UGIX)

We see UGIX as an 'intelligent front end' which can speak to all GIS in their own command languages yet which can be instructed by the user in a universally accepted and standard 'Geospeak'. A necessary preliminary to discussing UGIX, however, is to define a conceptual framework for the range of options available in GISs. This can be obtained from various different sources; table 1 summarises the findings of Rhind and Green (1988), and shows a superset of known functions, albeit described in general terms. GISWorld (1988) has provided a mapping from a similar list of basic functions to existing GIS (or, at least, what the vendors claim their

systems can achieve).

Essentially, the design for UGIX - the entire section within the main box - consists of three sub-systems, labelled A, B and C in figure 1. UGIX provides, in its most basic form, a command-level interface to (in theory) a variety of extant systems. In practice, we intend to build the prototype with only two system interfaces in the first instance. The contents of the boxes are :

Box A

The Decision Taker is the heart of UGIX. Upon receipt of commands or of output, it decides what resources and other information are needed to effect its tasks. Within it, are four sub-modules.

System Configuration module. This stores details of the UGIX configuration itself and of the systems with which it can communicate.

The Dialogue Handler. This handles all interaction between user and all sub-systems; thus it handles all input commands and also all reports (including graphic portrayals) resulting from these commands. It provides multiple methods of issuing commands but provides a 'point and click' default.

The Rules Table. This contains all the rules and protocols which control the actions of the Decision Taker.

The Mapper. This converts all the UGIX commands into those required to drive the other systems (where this is possible given their capabilities) and converts all output from them back into UGIX reports.

Box B

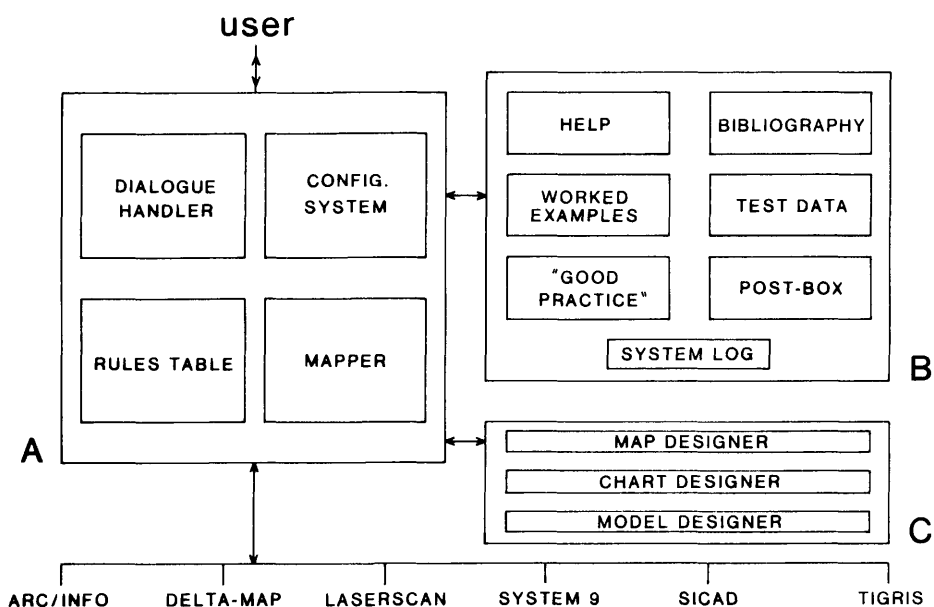
This consists of an enhanced version of GIST (Raper and Green 1989). In it is a sub-module for providing Help facilities (in principle in regard to all the systems with which it UGIX is linked, as well as the host system itself and GIS concepts, a glossary of terms, etc). Other sub-modules provide worked examples, a description of 'good practice' in carrying out the types of analysis being requested, some test data sets, a bibliography searchable on key worded topics, a log of all actions carried out by the system and the results and a post box for leaving messages (over the network if this is set up) for the postmaster on problems encountered. The messages may be initiated either by the user or by UGIX.

Box C

This module is the lowest of our priorities at present. It is intended to add 'intelligence' to UGIX's capabilities. We envisage at least three sub-modules, each with a suffix of 'designer'. Thus one sub-module provides not only within UGIX definition of scaling,

clipping, feature selection and symbolism but also more 'intelligent' input such as context-setting (Egenhofer and Frank 1988); its capabilities approximate to those of the 'ideal mapping package' (Blatchford and Rhind 1989). Another sub-module provides a similar (though simpler) function for design of histograms, graphs, scatter plots, etc and the third is an interactive model design facility which permits the specification and calibration of user-specified models.

Figure 1 : functioning parts of UGIX



Many of these facilities already exist in one form or another. Thus far, we know of no situation where they have been brought together in a form approximating to UGIX. Our initial intention is to regard UGIX only as a language translator between 'UGIX-speak' and the specific vocabularies, grammars and other conventions used by different systems. In the longer term, we have greater ambitions for UGIX : we would wish to provide an option whereby users could instruct it in any of the languages used by any of the 'recognised' systems and drive any one of the others. This presupposes, of course, that the concepts in each system are mappable to each other system. Where this is not the case, the Rules Table and the Mapper will ensure that no attempt is made carry out the impossible. Finally, we would wish to use UGIX as a data translation tool for those circumstances where data had to be translated from one system's representation into another.

CONCLUSIONS

UGIX exists only in parts at present. To make it work as outlined will require resources in excess of those currently available to us if it is to be made available over a time scale of a year or two. We believe - and have good evidence to support the belief - that it could be built most speedily using Hypercard and other facilities for MacIntosh computers, especially if allied to tools such as SequeLink. We do not, of course, underestimate the task: it will, for instance, involve keeping up - to - date with new developments in different GIS systems. For this reason, we would be happy to embark upon UGIX as a collaborative venture, especially with those individuals or groups who have experience of systems of which we have no expert knowledge.

Table 1. A classification of GIS functions needed for UGIX

Data Input and Encoding

Data capture (eg. manual or automatic digitising)

Data validation and editing (eg. quality checking)

Data storage and structuring (eg. construction of link/node topology, chain coding, etc.)

Data Manipulation

Structure conversion (eg. vector-to-raster, quadtrees to vector)

Geometric conversion (algebraic and 'rubbersheet')

Generalisation and classification

Enhancement (eg. edge enhancement, line fractalisation)

Abstraction (eg. calculation of centroids, Thiessen polygons)

Data Retrieval

Selective retrieval of information based on spatial or thematic criteria, including 'browse' facilities.

Data Analysis

Spatial analysis (eg. polygon overlay, route allocation, intervisibility, slope and aspect calculation)

Statistical analysis (eg. frequency analysis, dispersion)

Measurement (of lines, areas, volumes, distance, direction)

Data Display

Graphical display of maps, graphs, etc. on both graphical display and on hard copy devices.

Report writing and progress messaging

Database Management

Integrated database management facilities include: support and monitoring of multi-user access to the databases; provision of 'roll-back' facilities for use in the event of system failure; organisation of the database for efficient storage and retrieval without data redundancy; automatic maintenance of database security and integrity; providing the user with a 'data-independent' view of the database.

Note: This classification is derived from that of Rhind and Green (1988) and that in turn was based partly on the work of various other authors (see their paper for details)

REFERENCES

- ACA (1988) The Proposed Standard for Digital Cartographic Data. American Cartographer, 15, 1, 137pp.
- Blatchford R. and Rhind D.W. (1989) The ideal mapping package. In Rhind D.W. and Taylor D.R.F. (eds) Cartography - yesterday, today and tomorrow, Elsevier, Amsterdam
- Campbell W.J. (1977) Computer algorithms for spatial data. Area, 9-2, 106-8.
- Egenhofer M.J. and Frank A.U. (1988) Designing object-oriented query languages for GIS : human interface aspects. Proc. Third Intl. Sympos. on Spatial Data Handling, 79-96, Intl. Geographical Union, Sydney.
- GISWorld (1988) GIS Software Survey. GIS World, July, 7-11
- Goh, Pong-Chai (forthcoming) A graphic query language for cartographic and land information systems. Intl. J. of GIS
- Nash S.H. and Redwine S.T. (1987) A map of the world of software-related standards, guidelines and related practices. Computer Standards and Interfaces, 6.
- OS (1987) The National Transfer Format, Ordnance Survey, Southampton, UK.
- Palmer B.L. and Frank A.U. (1988) Spatial languages. Proc. Third Intl. Sympos. on Spatial Data Handling, 201-9, Intl. Geographical Union, Sydney
- Raper J.F. and Green N.P.A. (1989) GIST : an object-oriented approach to a GIS Tutor. Proc. Auto Carto 9
- Rhind D.W. (1981) Geographic Information Systems in Britain. In N.Wrigley and R.J.Bennett Quantitative Geography, 17-35, Routledge and Kegan Paul, London.
- Rhind D.W. (1987) Recent developments in GIS in Britain. Intl. J. of GIS, 1, 3, 229-41.
- Rhind D.W., Armstrong P.A. and Openshaw S. (1988) The Domesday machine : a nationwide GIS, Geogr. Jl., 154, 1, 56-68.
- Rhind D.W. and Green N.P.A. (1988) Design of a GIS for a heterogeneous scientific community. Intl. J. of GIS, 2, 2, 171-89.

THE SOUTH AFRICAN STANDARD FOR THE EXCHANGE OF DIGITAL GEO-REFERENCED INFORMATION

A K Cooper, Project Leader: GIS
Centre for Advanced Computing and Decision Support,
CSIR, P O Box 395, PRETORIA, 0001,
Republic of South Africa

BIOGRAPHICAL SKETCH

Antony Cooper holds the BSc (Information Processing) and BSc (Honours) in Computer Science degrees from Rhodes University, Grahamstown. He is a recipient of a RICS/Auto Carto London Education Trust Award and is a member of the Council of the South African Institute of Computer Scientists. His research interests include standards for the exchange of digital geographically referenced information, fundamental concepts of digital geographically referenced information, digital terrain models and applications thereof.

ABSTRACT

Geographically referenced (geo-referenced) information consists of all information that refers to the human-environment system and that can be localized in space and time. This includes cadastral, topographic, hydrographic and statistical information. The need for standards for the exchange of digital geo-referenced information is well known. The author was a member of the project team which drafted the South African standard and is a member of the committee charged with maintaining this standard.

This paper will provide a technical overview of the South African standard for the exchange of digital geo-referenced information. It will describe briefly our concepts of geo-referenced information and the relational model used, which makes the standard easy to use and update. A set of data being exchanged consists of the *File Identification* (a fixed length, fixed format file that identifies the data), the *Global Information Section* (giving general details about the data being exchanged, such as reference surface and coordinate offsets used) and the *Geo-referenced Information Relations* (containing the data being exchanged). This paper will describe these components, specifying how they cater for information on data quality, classification, non-spatial attributes, alternate spatial attributes, vector and raster data.

INTRODUCTION

Geographically referenced (geo-referenced) information consists of all information that refers to the human-environment system and that can be localized in space and time. Thus, geo-referenced information is of a diverse nature and includes cadastral, topographic, hydrographic, geological, remotely sensed and statistical information. In a digital form, geo-referenced information consists of vector, raster and alphanumeric data, as well as the inter-relationships between the various data. Standards for the exchange of digital geo-referenced information have to cater for the diversity in the nature of the digital data and the diversity in the nature of the geo-referenced information.

This paper describes the South African standard [Clarke *et al* 1987], which attempts to cater for all forms of digital geo-referenced information. The standard

is based on a relational model, which makes it modular and thus flexible and relatively easy to use and update. A set of data being exchanged consists of a File Identification, a Global Information Section and a number of Geo-referenced Information Relations.

The standard has been reviewed by Lane [1988].

THE NATURE OF GEO-REFERENCED INFORMATION

Digital geo-referenced information is a representation of part of the real world and typically its location in space and time is recorded in two or even three spatial dimensions (typically the two planimetric dimensions and the vertical distance above, or below, some reference surface) — only rarely is its location recorded in the temporal dimension. The current version of the exchange standard caters for two and three dimensions. There are three forms of digital geo-referenced information, namely *vector*, *raster* and *alphanumeric*. In addition, there is information on the spatial relationships inherent in the data, namely the *topology*. The exchange standard provides for the above, as well as mechanisms for exchanging information on the quality of the digital data and *alternate spatial attributes* — multiple versions of the digital representation of an entity.

Features

Features are the basic entities of digital geo-referenced information. A simple feature is a set of one or more uniquely identifiable objects in the real world where the defined characteristics of the objects are consistent throughout all the objects. Features can be man-made or natural, real or abstract. These defined characteristics are known as the *attributes* of the features, and can be *spatial* (that is, dependent on the feature's position in the n-dimensional space) or *non-spatial* (that is, independent of the feature's position — also known as the descriptive information of the feature). Thus, descriptive geo-referenced information is fixed in time and space through the features.

Classification is the arrangement of features into classes or groups and should be done on the basis of the *qualitative* characteristics of the objects, such as their function, and not on their *quantitative* characteristics. A feature's classification should be based on those of its characteristics that are least likely to change. There is a fine distinction between the non-spatial attributes of a feature and its classification because for different users, different criteria for classifying the information apply. One could even consider the classification itself to be a non-spatial attribute [Cooper 1987a].

While the exchange standard may be used with any classification scheme, the standard includes a skeleton classification scheme based on a variable-level hierarchical model for classification [Clarke *et al* 1987, Scheepers *et al* 1986].

Spatial attributes

A *spatial attribute* is an attribute whose value is a subset of any n-dimensional space — this version of the exchange standard caters for only two and three dimensions as they are the most commonly used. Should further dimensions become widely used, the standard will be expanded to cater for them, which should not prove difficult. Note that in the current version of the exchange standard, temporal values may still be recorded as non-spatial attributes. Spatial attributes may be *vector* (that is, positional data recorded as coordinate tuples forming nodes, chains, etc) or *raster* (that is, data expressed as a tessellation of cells, with spatial position implicit in the ordering of the cells).

The four fundamental types of two-dimensional vector spatial attributes are *nodes*, *chains*, *arcs* and *regions*, while the fundamental raster spatial attribute is the *matrix*.

A *node* is a 0-dimensional object with an n-tuple of coordinates specifying its position in n-dimensional space. The position of a *point feature* is described by a single node.

A *chain* is an ordered undirected sequence of n-tuples of coordinates with a node at each end. An *arc* is any continuous part of the circumference of a circle with a node at each end. The position of a *line feature* is described by a set of one or more chains and/or arcs, which do not necessarily form a continuous object.

A *region* is the interior of a continuous and closed sequence of one or more chains and/or arcs, known as the region's outer boundary. The position of an *area feature* is described by a set of one or more regions, which do not necessarily form a continuous object.

A *matrix* consists of an n-tuple of coordinates, specifying its origin, and an m-dimensional rectangular tessellation of data values encoded in a pre-defined format. The position of a *grid feature* is described by a set of one or more matrices, which do not necessarily form a continuous object.

Compound features are those which consist of one or more other features. This allows the user to build a hierarchy of features, for those occasions when the individual constituent features have their own non-spatial attributes (and classification), but together they have other additional non-spatial attributes and a classification.

Topology

The exchange standard caters for two topological relationships, namely *coincidence* and *exclusion*. Coincidence refers to the sharing of common sets of coordinate tuples, and is modelled by having more than one feature share the same spatial attributes. Exclusion refers to area features that consist of regions that wholly contain other regions that do not form a part of the area feature. Exclusion is catered for explicitly through two relations in the exchange standard.

Alternate spatial attributes

A feature has *alternate spatial attributes* when it is represented by a number of different sets of spatial attributes, where each set defines fully the location of the feature. An *alternate spatial attribute scheme* determines the manner in which the different alternate spatial attributes are related to their features. There are two main reasons as to why a feature would have alternate spatial attributes.

Firstly, in an area with a high density of features, the graphical representation of the area (be it on a computer screen or hard copy) would be messy, unless the display of some of the features could be suppressed, or unless some of them could be represented in a simplified manner. However, for analysis on the spatial attributes of the features, one would prefer to retain the spatial attributes of all the features and to as much detail as possible. Alternate spatial attributes allow one to keep different versions of the spatial attributes for the features to solve this problem — at one level, the alternate spatial attributes are for display, while at another level they are for analysis.

Secondly, if one deals with data at greatly disparate scales, one would like to retain different, scale dependent, versions of the spatial attributes of those features which appear at both small and large scales — automatic generalization of spatial data from a large scale to a small scale is still an interesting research area, and it is not possible to create large scale spatial data from small scale data! Again, alternate spatial attributes allow one to keep more than one set of spatial attributes for a feature.

In the exchange standard, an entry in the Global Information Section determines

whether alternate spatial attributes are used in the data set being exchanged, and if so, which scheme is used. If they are used, then in the Geo-referenced Information Relations, the field *Alternate spatial attribute* is used in every relation between features and spatial attributes, as well as in the two relations which define the type of the feature (point, line, etc) and its planimetric spatial domain. If alternate spatial attributes are not used, then the field is ignored completely.

There is a relation in the Geo-referenced Information Relations for exchanging, with the data set, an alternate spatial attribute scheme — no such scheme is defined in the current version of the exchange standard.

Information on the quality of the digital data

The American National Committee for Digital Cartographic Data Standards (NCDCDS) identified the nature of information on the quality of digital geo-referenced information, and which information should be recorded [Moellering 1986, Chrisman 1986].

Although some exchange standards allow for the encoding of some forms of information on the quality the digital data, such as the British standard [Sowton & Haywood 1987], we have followed the lead of the NCDCDS and allow the information on quality to be exchanged as free text only. A relation is used which may be included as often as necessary in amongst the Geo-referenced Information Relations. The granularity of the information on quality can thus vary from coarse (referring to the whole data set) to fine (referring to a section containing only one instance of a particular relation) [Cooper 1987a].

Only once the quantification of information on the quality of digital geo-referenced information is well understood, will the exchange standard address the encoding of such information on the quality of the digital data.

THE RELATIONAL MODEL OF THE EXCHANGE STANDARD

A data set in the format of an exchange standard is not a data base — it is merely a set of data that has been extracted from one data base with the purpose of being incorporated into another data base. To be successful, an exchange standard must be independent of the data bases that might be interfaced to it.

There are three common models for data structures, namely the hierarchical, the network and the relational. This exchange standard uses a relational model because it is inherently modular and more flexible than the hierarchical or network models. In a relational structure, the data are represented in a single uniform manner, and thus operations on the data are robust and simple to implement.

When creating a data set in the format of the exchange standard, one merely omits those relations for which one has no data. It is easy to add new relations to the exchange standard — in fact, data that can be exchanged through the relational structure of the exchange standard should always be able to be exchanged through the exchange standard, no matter how many new relations are added to cater for new concepts or types of data [Cooper 1987b]. This is achieved by adding new relations and leaving the existing ones as they are, rather than modifying the existing relations.

It is desirable to have a degree of normalization in data in a relational form [Van Roessel 1987]. There are some relations in the exchange standard for which normalization was not really feasible due to the excessive storage and processing overheads that would be introduced. For example, the records in the relation containing the internal coordinates of chains have variable numbers of fields (one field for each coordinate). For the rest of the relations, an attempt was made to normalize the relations to the third normal form. This required the introduction

of a *Sequence number* field to the keys of those relations where the keys were not unique, for example the relation relating classification to feature — any feature class may have many features with that classification. However, the sequence number appears only in the document describing the standard and not in the data being exchanged. As the data in the data set have an inherent ordering, the sequence number is implied by the record's position in the data set.

As an example, the following are the relations which relate an area feature to its classification and its spatial attributes:

1. *Feature/classification* which relates:
 $Feature\ ID \iff Classification$
2. *Feature/feature type* which relates:
 $Feature\ ID \iff Feature\ type$
3. *Area feature/included regions* which relates:
 $Feature\ ID \iff Region\ ID$
4. *Region/chains & arcs & direction* which relates:
 $Region\ ID \iff Indication\ of\ chain\ or\ arc \cup Chain\ ID \cup Direction\ indicator$
5. *Chain/nodes & coordinate tuples* which relates:
 $Chain\ ID \iff Node\ ID \cup Node\ ID \cup Length\ of\ chain \cup Data\ ID$
6. *Node/coordinate tuple* which relates:
 $Node\ ID \iff Coordinate\ tuple$
7. *Chain data* which relates:
 $Data\ ID \iff Coordinate\ tuples$

Relation 1 classifies the feature, relation 2 identifies the feature as an area feature, relation 3 connects the feature to its region spatial attribute, relation 4 performs the topological link between the region and the chains and arcs which form its boundary (specifying whether the chains and arcs are used forwards or backwards), relation 5 links the chains to their start and end nodes and to their internal coordinate tuples, relation 6 specifies the coordinate tuples identifying the locations of the nodes and relation 7 contains all the internal coordinate tuples for the chains.

FILE IDENTIFICATION

The *File Identification* is a fixed format file for identifying the set of data being exchanged. It is 2048 bytes long and consists of standard 7-bit ASCII characters. The fixed format facilitates the extraction of the various fields, both by computers and humans! Most of the information in the File Identification is in a free text, human-readable form (for example, the *Data identification*, *Source* and *Maintenance organizations*, *Copyright statement* and *Comments*), while some is in a formatted, computer-readable form, yet still intelligible to a human (for example, the *Volume number*, *Time* and *Date stamps*, *Physical record size* and *Blocking factor*).

The purpose of the File Identification is to allow the recipient of the data set to identify the data set, its currency and its relevance to his geographical information system, without having to do involved interpretation of the data set. The volumes of digital geo-referenced information that any user might receive, and thus the volumes of various physical media containing such information that might reside in the user's storage, are potentially enormous. The File Identification is there

to provide identification of the data should the physical label on the media prove to be missing, illegible or cryptic.

In addition, the File Identification provides some information to the interface program attempting to interpret the data set — for example, the *Physical record size* and *Blocking factor* indicate the manner in which the data are stored on the physical exchange medium, and the *ASCII/Binary* and *Explicit lengths/Delimiters* flags indicate whether the data are stored using 7-bit ASCII characters or in binary, and whether the fields are separated by delimiters of whether the lengths of the fields are determined by explicit length fields appearing before each field.

The File Identification forms the *first physical file* of a data set being exchanged. The rest of the data forms the *second physical file*. On a magnetic tape, these two files are separated by two end-of-file markers. The first version of the exchange standard describes the use of only magnetic tape as the physical exchange medium, as very few users in South Africa use anything else at this stage. This does not preclude the use of any other exchange medium, however.

GLOBAL INFORMATION SECTION

The *Global Information Section* provides details of the data being exchanged, such as the *Projection or coordinate system* and the *Reference surface* used. Some consider this information to be information on the quality of the data being exchanged — we consider the information to be critical for the correct interpretation of the data being exchanged.

The entries in the Global Information Section consist of variable length fields and records with either delimiters between the fields and records, or with explicit lengths at the beginning of each field, as indicated in the File Identification. However, the use of delimiters is recommended as they are conceptually easier to understand and implement, both when creating and interpreting the data set.

Most of the entries have default values and are thus optional. Those that do not have defaults are essential, for example the *Standard meridians & parallels & scale factor*.

Other entries in the Global Information Section include the *Units* and *Increment* of the *Planimetric* and *Vertical coordinate resolutions*, the *Bounding planimetric quadrilateral coordinate tuples* and the *Data quality, Feature classification, Attribute* and *Alternate spatial attribute schemes* and *release numbers*.

GEO-REFERENCED INFORMATION RELATIONS

The *Geo-referenced Information Relations* contain the actual data being exchanged. Each section, which corresponds to a table in a relational data base, contains a sequence of instances of a particular relation.

As in the Global Information Section, the sections in the Geo-referenced Information Relations consist of variable length fields and records with either delimiters between the fields and records (and sections), or with explicit lengths at the beginning of each field, as indicated in the File Identification. In addition, there is a relation, namely *TEMPLATE*, which allows the creator of the data set the option of using explicit lengths to set up templates for the fields, and hence make the fields fixed length fields. However, the use of delimiters is recommended.

The relation for exchanging information on the quality of the digital data, namely *DATAQUAL*, consists of free text which describes the quality of the data. In addition, the *Description* fields in the relations for exchanging the classification,

namely *EXCHCLAS*, the non-spatial attribute scheme, namely *EXCHATTR*, and the alternate spatial attribute scheme, namely *EXCHASAS*, also contain free text. All other fields and relations contain information in a format encoded explicitly for automatic interpretation by the interface program of the recipient.

EXCHCLAS and *EXCHATTR* provide the user with a data dictionary facility for exchanging the definitions of the classification and attribute schemes with the data set.

A number of relations have inverse relations. For example, there is the relation *FEATCLAS* relating a feature to its classification, and the inverse relation *CLASFEAT*, relating a feature class to features with that classification. All the relations between features and spatial attributes have inverses, for example *FEATNODE* (relating point features to nodes) and *NODEFEAT* (relating nodes to point features), and all the topological relations amongst the spatial attributes, for example *REGICHAI* (relating regions to chains and arcs) and *CHAIREGI* (relating chains and arcs to regions).

Finally, there are the geometric data relations which contain the coordinate tuples and constitute the bulk of the data set — especially *CHAI*DATA, which contains the internal coordinate tuples of the chains.

DIFFERENCES WITH RESPECT TO OTHER STANDARDS

The designers of the South African national standard for the exchange of digital geo-referenced information had the benefit of drawing on the work performed in other countries on similar exchange standards, as well as the opportunity of holding discussions with some of the designers of these standards — in particular, the standards of Australia [SAA 1981], North America [DCDSTF 1988], United Kingdom [Sowton & Haywood 1987] and the International Hydrographic Organization (IHO) [CEDD 1986].

We believe that the South African standard is the first to attempt to cater for all forms of digital geo-referenced information — the other standards are generally targeted at either cartographic or hydrographic information.

The Australian standard uses a hierarchical structure, the British standard uses a combination of network and relational models, the American standard allows the use of either a hierarchical or a relational model and the IHO standard uses a network model. The South African standard uses a relational model.

All four of the abovementioned standards include a full classification scheme and the American, British and IHO's standards include comprehensive non-spatial attribute schemes. The American, British and IHO's standards allow the use of any classification and non-spatial attribute schemes. The South African standard includes a skeleton classification and non-spatial attribute scheme, and allows the use of any such scheme.

CONCLUSIONS

While the community in South Africa supports the national exchange standard in principle, few attempts have been made to implement it. The Institute for Natural Resources at the University of Natal in Pietermaritzburg have implemented a significant subset of the interface in both directions between their home-grown geographical information system (GIS) and the exchange standard, and other organizations are at the design stage of the implementation. For their tender for a GIS, the Department of Water Affairs distributed benchmark data in the format of the exchange standard [Olivier *et al* 1989]. These efforts have shown that the

basic concept of the exchange standard is sound. They have also highlighted a few problems with some of the relations in the exchange standard. None of these problems is critical and they will be addressed in the next edition of the exchange standard, due to be published in the first half of 1989. In addition, they have unearthed some interesting problems concerning the fundamental nature of digital geo-referenced information [Greenwood 1988].

In addition to maintaining the exchange standard, the National Exchange Standard Committee will keep a record of digital geo-referenced information available in South Africa. To this end, a questionnaire was distributed [NESC 1988].

We believe that the process of developing this standard has made a significant contribution to creating more awareness among the South African GIS community of the fundamental concepts of geo-referenced information [Cooper 1988].

ACKNOWLEDGEMENTS

I should like to thank the Trustees of the Education Trust of The Royal Institution of Chartered Surveyors (RICS) for awarding me a RICS/Auto Carto London Educational Trust Award, which made presentation of this paper possible.

I should like to thank the other members of the project team, Derek Clarke, Hester van Rooyen and Elri Liebenberg, my colleagues at the Centre for Advanced Computing and Decision Support and the GIS community, both in South Africa and abroad, for all their help, criticisms and advice.

The opinions expressed in this paper are those of the author and not necessarily of the CSIR.

REFERENCES

- Chrisman NR, September 1986, *Obtaining information on quality of digital data*, Proceedings: Auto Carto London, Vol 1, pp 350-358.
- Clarke DG, Cooper AK, Liebenberg EC & Van Rooyen MH, September 1987, *A national standard for the exchange of digital geo-referenced information*, NRIMS CSIR Special Report SWISK 45, 201 pp.
- Committee on the Exchange of Digital Data, November 1986, *Format for the exchange of digital hydrographic data*, International Hydrographic Organization, 350 pp.
- Cooper AK, March 1987, *Thoughts on exchanging geographical information*, Proceedings: 1987 ASPRS-ACSM Annual Convention, Vol 5, pp 1-9.
- Cooper AK, June 1987, *A data structure for exchanging geographical information*, Proceedings: 4th South African Computer Symposium, pp 267-277.
- Cooper AK, September 1988, *Exchanging geographically referenced information — a status report*, Proceedings: Computer Graphics '88, pp B1-6 - B1-20.
- Digital Cartographic Data Standards Task Force, January 1988, *The proposed standard for digital cartographic data*, American Cartographer, Vol 16, No 1.
- Greenwood PH, September 1988, *Using the proposed national exchange standard for GIS data*, Proceedings: Computer Graphics '88, pp B1-21 - B1-29.

- Lane A, 1988, *Review: a national standard for the exchange of digital referenced information*, International Journal of Geographical Information Systems, Vol 2, No 1, pp 81–82.
- Moellering H, ed, January 1985, *Digital cartographic data standards: an interim proposed standard*, Report no 6, National Committee for Digital Cartographic Data Standards, 164 pp.
- National Exchange Standard Committee, July 1988, *Questionnaire: National Exchange Standard*, Chief Directorate: Surveys and Mapping, 11 pp.
- Olivier JJ, Greenwood PH, Cooper AK, McPherson DR & Engelbrecht R, April 1989, *Selecting a GIS*, Proceedings: Auto Carto 9.
- Scheepers CF, Van Biljon WR & Cooper AK, September 1986, *Guidelines to set up a classification for geographical information*, NRIMS CSIR Internal Report I723, 12 pp.
- Sowton M & Haywood P, chair, January 1987, *The national transfer format: release 1.0*, Ordnance Survey, Southampton.
- Standards Association of Australia, August 1981, *Interchange of feature coded digital mapping data*, Australian Standard 2482–1981, 24 pp.
- Van Roessel JW, 1987, *Design of a spatial data structure using the relational normal forms*, International Journal of Geographical Information Systems, Vol 1, No 1, pp 33–50.

THE TELECOMMUNICATION OF MAP AND CHART DATA

T. Evangelatos Canadian Hydrographic Service, Ottawa	Z. Jiwani Energy, Mines and Resources, Ottawa	D. McKellar Department of National Defence Canada	C.D. O'Brien IDON Corporation Ottawa
--	---	--	--

ABSTRACT

The structuring and communication of electronically formatted information is becoming increasingly important in a wide number of diverse fields. Industrial and office information systems are becoming widespread and world standards are developing both for the establishment of data communications networks as well as for the structuring of data to be communicated over these networks. A major effort is going on world wide for the development of a suite of international standards for Open Systems Interconnection (OSI) which form the basis of the international data communications networks. Standards for the structuring of data are developing for Facsimile, Computer Graphics, Computer Aided Design and Manufacturing, Office Automation (Forms), Videotex (Home Information Services) as well as for Cartography and Navigation Systems.

This paper discusses the trends in telecommunications standards and shows how they apply to the communications of cartographic information. The relationship of the International Standards Organization standards ISO 8824/5 (Abstract Syntax Notation and Coding), ISO 8211 (Data Descriptive File Format) and ISO 2022 (Code Extension) to cartographic data interchange is described. The Map And Chart Data Interchange Format (MACDIF) is discussed as a method of interchanging cartographic information in a flexible manner compatible with developing telecommunications networks.

1.0 INTRODUCTION

Tremendous changes are occurring in the fields of cartography and hydrography and although these disciplines have used computers for many years, it is only recently that the direct interchange and communication of digital data has been undertaken. With the need for data communications, the requirement for standards has become more important. The early efforts towards standardization concentrated on the structuring of data, but was limited to traditional magnetic tape interchange. Current efforts are also addressing the need for communicating data over public telecommunication networks based upon international telecommunication standards.

The effort to develop MACDIF is being carried on cooperatively by several Canadian Federal Departments, in particular the Canadian Hydrographic Service, the Department of Energy Mines and Resources, the Department of National Defence, the Public Archives of Canada and the Department of Communications as well as with the Ontario Ministry of Natural Resources, and the U.S. National Ocean Service.

MACDIF is a flexible format which can be used to communicate anything from raw digitized map information to a fully symbolized and cartographically enhanced map or chart. Annotation may be in English, French or any other language. MACDIF organizes information into a number of categories which define the overall structure of the spatial data, its relation to a world coordinate system, the features which make up the data set, their attributes and boundaries, and optionally any related symbolization and topological relationships. This data format allows for a blind interchange; that is, there is only one single flexible format for encoding data which may be interpreted by various levels of receiving computer or terminal devices.

The rapid establishment of telecommunication facilities has permitted the direct communication of digital information. This is leading to more sophisticated formats for the

representation, storage and communication of spatial data. In addition to creating standards for the organization of the underlying information content, it is also important that the data be formatted in such a manner so that it can be communicated over a variety of networks in a manner independent of the supporting telecommunications media.

Standards are being developed in the major international standardization bodies which permit the interconnection of virtually any digital data communications system. A new standard is required for the representation of spatial data which builds upon these data communications standards and defines a method of organizing the data in a flexible manner so that all or some of it may be used in a variety of applications, ranging from the production of paper maps and charts, to electronic display and non presentation uses of the data.

2.0 GENERAL CONCEPT

The establishment of a standard format for encoding map and chart data for the purpose of interchange and storage will have a profound impact. Not only will it be possible to communicate the electronic replacement of paper maps or charts, but a wide range of applications may be developed which make use of the attributes pertaining to map or chart features. The proposed interchange format provides the necessary common reference point to unlock the growing store of mapping and charting information and so facilitate the utilization of that information in a large number of diverse applications and by a broad audience. MACDIF accommodates the requirements of interchanging map and chart data between agencies and for the distribution of such data to private, commercial, and public users.

MACDIF can be viewed in different ways dependent upon the context in which it is discussed. From the technical point of view it is a coding scheme for the representation and communication of map and chart data; that is, it is a set of rules, a grammar, by which one may represent (encode) a digital description of a map or chart. The information is structured according to a rigorous, unambiguous syntax. This establishes a norm which forms the basis upon which to build a number of different independent applications, all sharing common data. This development of a general underlying coding scheme promotes the compatible development of a number of broadly-based applications.

MACDIF is also termed a proposed standard since it is intended that it be used as the common basis for a number of applications. The term standard is often misused to represent the specification of a commonly used coding scheme or other specification upon which systems or applications are based. However, the term standard more rigorously applies to a norm which has been established in an open public forum so that it represents the consensus derived from the consolidated experience of the industry. The development of a standard is carried on under the auspices of a national or international standards-making body according to certain well-established formal procedures and adherence to the principles defined by other national and international standards. This also provides stability and a mechanism by which a standard may be publicly maintained and updated.

3.0 PRINCIPLES

MACDIF is designed to be a general standard for communicating spatial data and is intended both for professional use of mapping and charting agencies as well as for the dissemination of information to industry and the public in electronic form. In order to achieve the maximum flexibility, it was important that the coding scheme be designed according to certain universal principles. These principles include :

- independence from hardware constraints of the equipment used within the applications,
- independence from the media used for communications or storage,

- communications transmission efficiency,
- the ability to build upon other norms already established in the industry
- blind interchange,
- meaningful defaults,
- the ability to accommodate a wide variety of applications,
- the capability to support multiple languages (including non Roman scripts)
- the ability to accommodate modification and extension in a forward and, where possible, backward compatible manner, and
- stability by being a public domain standard.

In order to establish an independence from the communications network and the eventual database or presentation media, MACDIF is defined in an abstract manner. Only at a final stage in the processing of MACDIF data, received over a standardized communications facility, would a binding to a particular coordinate system be established. For example, by communicating MACDIF positional information as fractions of a normalized unit coordinate system, a device-independent rendering may be achieved for use in a database or for presentation on anydisplay screen or on any plotter. That is, the coordinate system used within MACDIF is based on a unit square (with a 0 to 1 range for X and Y). Parameters for a transformation are communicated along with the data relating it to the real world coordinate system. Any presentation process may make use of the transformation specification and scale this data into its own device dependent coordinate system. This approach to coordinate specification is also the most efficient manner of storing and communicating such data since the number range matches exactly the area of interest and no extra digits are required to handle fixed biases. Coordinate and other information is packed into a small number of bytes while retaining the capability to specify these values to various levels of precision.

Independence from the coordinate system used in the target display device is not the only dimension of independence which is required. All parameterized variables associated with the graphical presentation of map or chart information should be specified in terms of normalized variables where possible.

The MACDIF coding scheme is also structured so as to be independent of the manner by which it is communicated or stored. The format is based on existing telecommunications standards so that data may be communicated over public telecommunications facilities as well as specialized private facilities. In order to achieve this independence, MACDIF has been defined in alignment with the International Standards Organization's (ISO) reference model for Open Systems Interconnect (OSI); that is, it separates the coding and formatting of information from the means by which it is communicated. In OSI terms MACDIF is an application data format for general mapping and charting applications which utilizes and builds upon existing and specialized presentation data coding standards. MACDIF defines what the data entities are and makes use of existing or specialized presentation data coding standards in order to encode the data entities. For example, textual data is coded in terms of the International ASCII character code standard.

A principal feature of the MACDIF approach is the ability to support "blind interchange". This means that map and chart information is defined independent of context so that it may be communicated without the need for negotiations between the sending and receiving entities. The same format of data is used in all communication regardless of the application. The entity which receives the data interprets those portions of the data which pertain to a particular application. For example, an application which simply displays an outline map may not be interested in topological information, and would ignore it if it was encountered. On the other hand, "raw" data, as collected by a digitization/supplier organization, might

not contain sufficient information to support certain applications such as the production of a paper chart. This data, although in the same format, would require processing and cartographic enhancement before it could be used for such a purpose. It is important to avoid negotiation over the format of data in order to eliminate the need to reformat the data for communications with each type of receiving computer or terminal.

MACDIF provides facilities to encompass a broad range of applications such as structured and symbolized maps or thematic uses of maps and charts. It permits a comprehensive data description of a chart or map while maintaining flexibility and extensibility of the coding structure.

Multiple languages may be supported since the character sets used to encode textual information are drawn from the ISO 2375 registry of character sets. The basic character set is the International Reference Version (IRV) character code table, which is identical to ASCII (the American Standard Code for Information Interchange), except that it contains a generalized monetary symbol. Accented characters are handled by the use of a supplementary table of accents, diacritical marks and special characters. This table is standardized in ISO 6937 and used in several other international standards. By the combination of these two code tables characters for any Latin alphabet based language may be coded. Characters for other languages such as Hebrew, Greek, Cyrillic languages, Arabic languages, African languages and even Japanese and Chinese Kanji may also be invoked in this manner.

By basing MACDIF on existing standards for code extension, it benefits from the extensibility built into these standards. As new coded character sets are defined and included in the registry, they may be used within MACDIF. Similarly, the capability to support code extension permits MACDIF to take advantage of the advancements in the coding of graphical data currently standardized or under development in ISO.

4.0 MACDIF INFORMATION STRUCTURE

A map or chart is a highly structured description of a geographical area. A map consists of features such as lakes, rivers and roads which not only have a geographical boundary, but which also have a large number of other attributes. The number and type of features used in a map is unlimited and the volume of data required to describe a map or chart can be very large.

A data interchange format for the communication of mapping and charting information must be flexible in order to accommodate many different uses. These range from the communication of basic digitized map data from a supplier to the intercommunication of map or chart data extracted from a data base and transferred between agencies, or to the distribution of spatial information to industry or the public in electronic form. The data which is required to specify a map or a chart in each of these situations is somewhat different.

In order to provide flexibility of use, the Map And Chart Data Interchange Format (MACDIF) breaks the data description down into various sections. Each of these sections addresses a separate class of information in the map or chart description. Certain descriptors are mandatory since they contain basic map information, whereas other descriptors are optional since they contain auxiliary information which is required only in certain views of the spatial data

4.1 Overall Data Structure

The overall structure of a map or chart in MACDIF consists of a contiguous ordered unit of data. This application data unit can be broken down into an administrative header and data set descriptor. The header contains information such as the data set name, format version, update indicator etc. which are required to identify the map or chart in question. The map definition section contains the data describing the remaining components of a map or chart.

It contains a description of the transformation which positions the spatial data set into real world coordinates, a description of the features, attributes and boundary definitions which compose the data set and optional topological, symbolization or other related information. This is illustrated in Figure 1.

The overall map or chart structure is defined as a syntactic hierarchical tree where each section is broken into sub-sections and sub-sub-sections down to the data element. A MACDIF data file may contain several map descriptors under a single administrative header. A second map descriptor file may be used to overlay thematic information such as census population data over a basic geographic map extracted from a library.

The definition of a map is divided into several subsections each of which addresses a different portion of the information required to define a map or chart. These subsections are:

- Administrative Header Definition
- Transform Definition
- Feature Definition
- Segment Definition
- Topological Definition
- Symbolized Map Definition
- Associated Information Definition

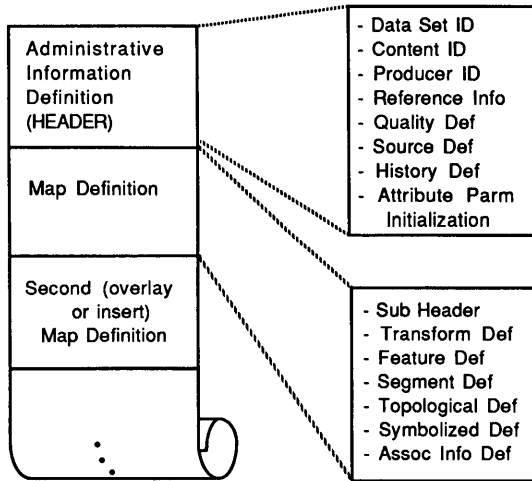


Figure 1 Overall Map (or Chart) Structure

5.0 RELATIONSHIP TO OPEN SYSTEMS INTERCONNECT (OSI)

The development of standards for telecommunications among computer based information processing systems is a central area of study in the international standardization committees. With the onset of the information economy throughout the world, it has become increasingly important to establish universal standards for communication. It is important that MACDIF coded information be available via any kind of data communications means. This section presents background information pertaining to telecommunications standards as they relate to MACDIF.

5.1 The OSI Communications Environment

The International Organization for Standardization (ISO) and the International Telephone and Telegraph Consultative Committee (CCITT), part of the United Nations-sponsored International Telegraphic Union (ITU), have been developing a general structure for a set of interlocking telecommunications standards which are expected in the near future to handle the majority of the world's data communications traffic. Underlying this work is a basic layered architectural model for Open Systems Interconnection (OSI). The principle behind this model is to separate the various operations involved in communicating data into seven independent layers. In concept, each layer is separate, and a number of different standards may be defined for each layer. For example, the lower layers might make use of different protocols for various communications media, such as over a terrestrial land line or over a satellite communication channel.

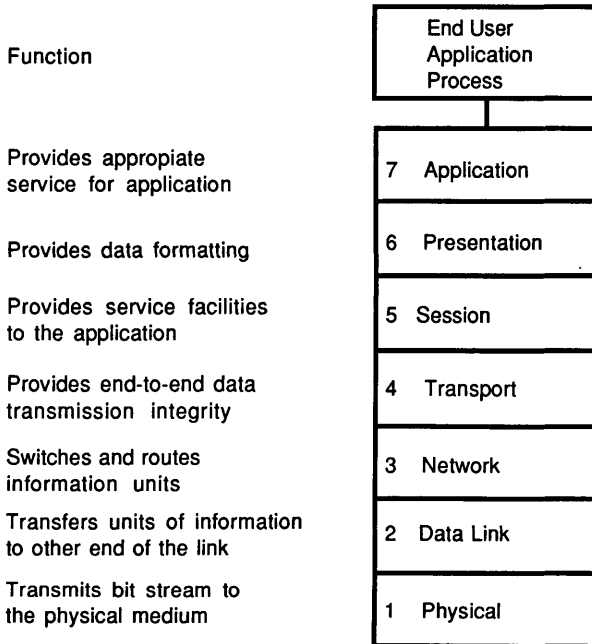


Figure 2: The Seven Functionally Separate Layers of the OSI Model

OSI is concerned with the communication of information between open systems. An open system can be any assemblage of communicating entities which is open to general interconnection. That is, the communications network is not predetermined and closed. The seven OSI layers shown in Figure 2 can be grouped into three broad categories. Layers 1 to 3 are concerned with communicating data over physical media such as wires, microwave links, satellite channels, etc. Layers 4 and 5 are concerned with the end-to-end communications dialogue and the assurance of maintaining end-to-end data integrity. Layers 6 and 7 are concerned with the data that is communicated. Layer 6, the presentation layer, defines the coded representation of the data and layer 7, the Application layer, provides resource management to the application process which makes use of the data.

Each of the layers of the OSI model is independent and different standards may be substituted at each layer in different situations. For example, at the Presentation Layer (Layer 6), data may be coded in the ASCII character code standard in one situation or in

EBCDIC (an IBM standard) in another. The OSI reference model has recently been adopted as an international standard. Along with this basic standard, ISO and CCITT are developing a number of specific standards for each layer which align with the model and which may be used to communicate over various communications media. Some of these protocols such as the X.25 packet-switched protocol for network communications are well established. Other protocols, such as the file and message handling protocols (X.400 series) for future electronic mail services, are under development.

The Presentation Layer (Layer 6) provide a means of encoding data and giving meaning to data entities. Data may be coded in alignment with any of a number of data coding standards such as ASCII, binary bit images, or other data syntaxes for graphical or other types of data. An Abstract Syntax Notation (ASN.1) has been defined in the standard ISO 8824 which permits data elements to be coded according to different data syntaxes for different types of data elements and assembled into an application-specific data format. Together with the Abstract Syntax Notation a set of encoding rules have been defined in ISO 8825 for communication over OSI data networks. The Application Layer (Layer 7) communicates directly with the End User Application Process. Specific Application Layer Service Element standards (SASE) are being developed for various applications. A set of Common Application Service Elements (CASE) is defined for all Application Layer standards for basic functions such as establishing a communication or recovery from an error condition.

For the communication of spatial information in an OSI environment, the communications facilities from the Session Level down can be assumed to be available. In addition, some standardized data syntaxes are available at the Presentation Layer and may be used to code data. At the Application Layer, a specific data format suited to the particular application of communicating map and chart data must be defined. To suit the OSI environment it should be specified in the Abstract Syntax Notation of ISO 8824.

5.2 Structure of the Interchange Format

The structure of the MACDIF syntax consists of a hierarchical tree of information elements. Each element is tagged so that it may be identified in a data set. The data set is interpreted by scanning the information in the order defined by the tree structured syntax. The process which interprets (or parses) the data set matches each data element in the data set to the type of data element expected by the syntax. For example an alphabetic code identifying a particular Feature Type is identified by a unique tag. This information element must be contained in the set of data following the numeric code identifying that particular Feature Number. Since each type of information element is identified by a unique tag, it is not necessary to include filler information in the data set for optional information which has not been specified.

This method of structuring the information content of MACDIF is extremely flexible and very efficient. The only overhead is the tag code on each information element, but even this minor overhead is offset by the fact that redundant information need not be included in the data set. There are no preset lengths for any information element since each element is delimited by the tag which identifies it. This provides total flexibility since information need only be specified to the precision which is meaningful. Redundant "zeros" or other place fillers are not required, and there is no limit to the length of a data element.

The general syntactic structure of MACDIF consists of an Administrative Header which identifies the Data Set and many overall parameters about the Map or Chart. This is followed by one or more Map Definition sections which contain data describing the remaining components of a Map or Chart. A portion of the description of MACDIF represented in ASN.1 format is presented below for illustrative purposes. Note that the complete description of MACDIF in ASN.1 is twenty-five pages long.

```
Digital-Map-or-Chart::= SEQUENCE { Map-Header,  
                                Map-Def-Section }
```

Map-Def-Section	::=	SEQUENCE OF	Map-Definition
Map-Definition	::=	SEQUENCE {	
		[1] Map-Sub-Header	OPTIONAL,
		[2] Transform-Definition-Section,	
		[3] Feature-Definition-Section,	
		[4] Segment-Definition-Section,	
		[5] Topological-Definition-Section	OPTIONAL,
		[6] Symbolization-Definition-Section	OPTIONAL,
		[7] Associated-Information-Definition-Section	OPTIONAL }
Map-Header	::=	SEQUENCE {	
		[1] Data-Set-ID,	
		[2] Content-ID,	
		[3] Producer-ID,	
		[4] Reference-Information,	
		[5] Quality-Declaration-Section	OPTIONAL,
		[6] Source-Declaration-Section	OPTIONAL,
		[7] History-Description	OPTIONAL,
		[8] Parameter-Definition	OPTIONAL }
Map-Sub-Header	::=	SEQUENCE {	
		[1] Content-ID	OPTIONAL,
		[2] Producer-ID	OPTIONAL,
		[3] Reference-Information	OPTIONAL,
		[4] Quality-Declaration-Section	OPTIONAL,
		[5] Source-Declaration-Section	OPTIONAL,
		[6] History-Description	OPTIONAL,
		[7] Parameter-Definition	OPTIONAL }

5.3 Data Formats

The Application layer of the MACDIF standard describes which information may be used in the description of a map or chart and how that information is interrelated. In addition, the Presentation layer specifies how MACDIF information would be represented in terms of a stream of digital data. This distinctly separate layer is responsible for the representation and coding of data. Both the Presentation and Application layers are also completely separate from the method by which the information is communicated or stored.

5.3.1 Supporting Data Syntaxes The information entities which make up the various components of a MACDIF specification of a map or chart consist of textual information, pictorial information, numerical parameters or specialized identifiers and pointers. Each of these must be represented as bit patterns in terms of various data syntaxes. For example, standards currently exist to encode the letters of the alphabet for almost all languages in the world. In North America the alphabetic coding standard is called ASCII. Similarly, standards exist or are under development for coding pictorial information such as graphical points, lines, and polygons and numerical and other information.

MACDIF makes use of existing standardized general data syntaxes to define textual and numerical formats in order to achieve maximum compatibility with telecommunications services and related applications defined in other contexts. Standards exist to define textual, numerical and pictorial data types, and since these types are defined by international standards they are implicit types for application by the ISO standard Abstract Syntax Notation in which the syntax of MACDIF is specified.

The coding of pictorial and numerical information in MACDIF is based on a standardized approach. MACDIF requires a pictorial data type primarily to communicate points, line segments, arcs and polygons in the definition of the boundary of features. Since boundary definitions are the principle data component in the definition of a map or chart, it is

important that the coding of pictorial coordinate data be compact and simple. Over eighty percent of the data volume of a map is pictorial. Complex drawing attributes are not required. For this reason only a simple pictorial coding scheme is needed by MACDIF.

5.4 Coding Standards

There are two principal methods of coding data: the bit-coded method and the byte-coded method. In the bit-coded method, meaning is given to specific bit patterns and all the bits in a bit stream are significant. In order for this method to work, a mechanism, usually provided as a service from an OSI lower layer, must be available in order to delimit the start of a bit sequence. This is slightly more efficient than the byte-coded method which is self-delimiting in that specific codes (Escape sequences) have been reserved to control the coding environment. However, the byte (or character)-coded method is more flexible. It does not rely on any services provided by the lower layers of a communications system.

One of the fundamental requirements for MACDIF is generality. Since data syntaxes defined according to the rules for character coding may be communicated both over bit-transparent OSI communications systems and character-oriented asynchronous protocols, MACDIF is built upon character coded data syntaxes. In the character coded approach eight-bit bytes (or octets) of data are organized into code tables where each code has an assigned meaning. Seven of the bits are used as an index into a 128 character code space and the eighth bit is reserved either for parity data error checking by lower layers of the communications system or for code extension to another code space of 128 characters.

Code tables may be invoked into the code space to specify the current interpretation for each of the respective codes. This is best illustrated by examining the ASCII code table. Individual characters from the Latin alphabet may be selected from the ASCII code table. An alternate code table of supplementary accents, diacritical marks and special characters may also be invoked into the code space to become the "In-Use Table". An accented character such as é in French may be composed by first selecting the non-spacing accent character from the supplementary code table and then selecting the character to be accented.

The international standard ISO 2375 establishes a world-wide registry of character code tables. Any characters from the repertoire of registered character sets may be invoked. The registry includes over 59 code tables which includes tables for the International Reference Version of the Latin alphabet (IRV), ASCII, Japanese Katakana, Greek, Cyrillic, Arabic, Supplementary Diacritical Marks, Accents and Special Characters, and the large Japanese and Chinese Kanji character sets. By incorporating this coding technique, MACDIF supports the presentation of text in virtually any written language.

The ISO 2022 standard on Code Extension in a Character-Coded Environment is the basic standard upon which a code table-oriented coding structure operates. It defines the procedures for code extension using the ESCAPE (ESC) character control function. A fixed set of rules allows all escape sequences to be interpreted as functions to manage the character coded environment or to be cleanly ignored if they do not apply. The ESCAPE character (code position 1/11) and the ESCAPE sequences are guaranteed to always have the same interpretation. MACDIF makes use of this standard as the underlying standard upon which the supporting data syntaxes used for coding text and pictorial data are based.

The use of code tables is not restricted to the coding of characters. Pictorial information and special control functions are also specified in this manner. The ISO working group on picture coding is establishing a standard approach to defining code tables for pictorial and numeric information. This is in support of the work on information publishing for telematic services under study in CCITT and computer graphics under study in ISO. This work forms the basis of the pictorial coding scheme used in MACDIF. The code table in MACDIF for pictorial primitives contains thirty-two primitives, such as commands to draw a point, a line, or a polygon, etc.

Another method of encoding data is the directory structured approach of ISO 8211. It is a standard designed for communicating structured data bases of information and is being

employed in exchanges using hard media, particularly in the United States. The ISO 8211 and the ISO 8824 (ASN) are both viable methods by which to encode spatial data. The ISO 8211 method more closely aligns with the practises used to establish data bases, whereas ISO 8824 aligns with telecommunication practices and is therefore more data efficient.

6.0 MACDIF DEVELOPMENT

The Map And Chart Data Interchange Format MACDIF is derived from work done by the Ontario Ministry of Natural Resources (OMNR) on the definition of a Map Data Interchange Format MDIF. The work on MDIF was done in conjunction with IDON Corporation and endeavoured to marry the needs of mapping to telecommunications. Cooperative work began in 1986 between the Canadian Hydrographic Service, other Canadian federal government departments (including the Department of Communication, Department of National Defence, Energy Mines and Resources, the Department of Supply and Services), the OMNR and the U.S. National Ocean Service in developing MACDIF. Work on the MDIF has been going on in parallel with the broader development of MACDIF, and MDIF can be considered as a profile of MACDIF for use in land mapping applications. The interest of the OMNR has been directed toward the acquisition and distribution of digital mapping information by a central mapping agency. Digital map data may be compiled by one or several industrial sources, and communicated to the OMNR by using MDIF. Through cooperation with industry in the process of reviewing and refining the interchange format, the OMNR is introducing MDIF as the method of interworking between the OMNR and the mapping information source suppliers in Ontario. MDIF will also be implemented for the distribution of digital mapping data to other agencies and to users of such data.

The interest of the Canadian Hydrographic Service is concentrated on the aspects of MACDIF which are more concerned with the "electronic chart", a concept in which digital navigational information can be made available on a display device on the bridge of a ship. MACDIF can play an important part in the distribution of electronic chart data and updates to the end users on vessels. CHS and most of the other agencies involved are also concerned about the broader issues of international standards for the telecommunications of digital map and chart information.

7.0 CONCLUSIONS

Although, at the moment, the demands for telecommunication based spatial data exchange standards are limited it seems apparent that such standards will become very important when:

1. There is a general availability of base mapping data.
2. A broader use of "Electronic" Atlases and similar digital publications develops - Initially these needs may be met with CD-ROMS, but in the longer term, applications will move on-line in order to provide users with up-to-date information.
3. There is wider use of electronic systems for navigation, both on land and water.
4. The GIS - Information Utility is developed.

MACDIF has the flexibility, efficiency, extensibility and other desirable capabilities to meet these evolving needs on both a national and international basis

REFERENCE

O'Brien, C.D. , 1988, Specification of the Map And Chart Data Interchange Format: - MACDIF, Unpublished Technical Report, IDON Corporation, Ottawa, Canada (Available from the Canadian Hydrographic Service, 615 Booth St., Ottawa, Canada, K1A0E6).

THE ESRC'S REGIONAL RESEARCH LABORATORIES :
AN ALTERNATIVE APPROACH TO THE NCGIA?

J.W. Shepherd¹, I. Masser², M. Blakemore³ and D.W. Rhind¹

- 1 : Department of Geography, Birkbeck College, University
of London, 7-15 Gresse Street, London W1P 1PA, UK
2 : Department of Town and Regional Planning,
University of Sheffield, UK
3 : Department of Geography, University of Durham, UK

ABSTRACT

The UK Economic and Social Research Council set up a pilot set of Regional Research Laboratories (RRLs) in early 1987. Following a successful review of this initiative, new RRLs have recently been set up and funded more intensively. The objectives of the RRLs include the need to engage in GIS research and teaching but also to provide data services, to carry out applications work (often in collaboration with users) and generally to proselytise on the capabilities and opportunities afforded by the technology. All have a regional orientation but many will also have some national focus.

The RRLs thus represent a somewhat different model to that set up under NSF funding and established at Santa Barbara, Buffalo and Maine (the NCGIA). The parallels and differences between these are set out, together with the lessons learned thus far in the RRLs and ESRC's future plans. The South East Regional Research Lab (SERRL), covering an area in which lives one third of Britain's population, is used as an example to illustrate the activities of the RRLs.

INTRODUCTION

The recent growth of interest in and commitment to Geographical Information Systems has already been well documented (e.g. Andersson 1987, Chen Shu-Peng 1987, Kubo 1987, Rhind 1987, Tomlinson 1987). This reflects a wider concern with the use of spatially referenced information to monitor, understand and (in some cases) manage both the natural environment and society itself. Many observers have pointed out, however, that much more research is required if we are to make use of such tools in a routine and efficient way.

The way in which different countries have focussed their research efforts differs. The national perception of priority areas and the scale of funding involved, the institutional context, the extent of the involvement of the private sector and the emphasis upon applied (as opposed to fundamental) research all vary between the plans of those countries of which we have knowledge. Perhaps the most advanced initiatives are :

- (i) the US National Centre for Geographic Information and Analysis (Abler 1987), funded by the National Science Foundation for up to eight years to the extent of \$10 million,

- (ii) the Dutch research consortium based on the University of Utrecht, the Technical University of Delft, the Agricultural University of Wageningen and the International Training Centre at Enschede and funded by the Netherlands Science Research Council for a four year period (Ottens 1988),
- (iii) the French activities, notably the creation of the Maison de la Geographie in Montpellier which has involved the creation of a research network linking 49 research teams across France, and
- (iv) the Regional Research Laboratory (RRL) programme in the UK.

The objective of this paper is to describe the background to the RRL initiative and to outline the research plans and some of the achievements to date of the Labs. This description is set in the context of other relevant developments in the UK and is illustrated by reference to the work of one of the RRLs - the South East Regional Research Lab (or SERRL). Finally, some comparisons are drawn with the US developments.

BACKGROUND TO THE RRLs

In the UK, funding for research work in universities arises from three main sources : as a component of central government's annual budget to universities (distributed through the Universities Funding Council (UFC) to individual institutions), as contract funds from research sponsors - increasingly from those in the private sector and multi-national organisations - and government funds distributed via the five Research Councils. In essence, the Research Councils and their parent body - the Advisory Board for the Research Councils - approximate to the US National Science Foundation. Thus the Agricultural and Food, the Economic and Social (ESRC), the Medical, the Natural Environment (NERC) and the Science and Engineering Research Councils all distribute money for research and for the support of post-graduate training, although the balance and total level of funding varies considerably between them. The ESRC is the smallest one, having an annual budget of about \$50 million.

The ESRC's initiative in setting up the RRLs can be considered to have arisen from two main sources, one internal and the other external. The first was a long - standing recognition by ESRC itself of the need to establish a suitable infrastructure for quantitative social science research, dating from 1967 when the Data (formerly the Survey) Archive was set up at the University of Essex. The primary concern of the Archive has progressively shifted over the years and is now very much upon secondary data sets, particularly those compiled by government agencies. As a result, it is now a national, multi-disciplinary facility which acquires, archives and disseminates machine-readable data sets to social science researchers and others. Its holdings of over 3,500 data sets make it the largest data archive of its type outside North America. In its role as

'data broker', the Archive forms part of the trend towards the commodification of information (Openshaw and Goddard 1987).

The more immediate internal trigger for ESRC to launch the RRL initiative was the findings of a joint ESRC/NSF committee. This committee isolated three topics which they saw as timely and central : Geographical Information Systems (GIS), election data bases and organisational data bases. Its findings included the recommendation that "...as a matter of social science policy, ESRC and NSF should maintain as a high priority the development of data resources that are national in scope, serve multiple objectives, are replicated all the time and are of continuing relevance to the respective research communities and to national goals" (ESRC/NSF 1986, p.5). Underlying this is a recognition that substantial investment in human skills is necessary if the value of new information technology is to be maximised. Establishing 'centres of excellence' and 'well found laboratories' along the lines of what exists in the natural sciences was seen as one way of achieving this goal.

The external 'trigger' was the efforts of the government's Committee of Enquiry into the Handling of Geographic Information - even before publication of the final report (DoE 1987) of this, the Chorley Committee. The report made a strong case for additional research and education in the GIS field and urged a commitment by ESRC and NERC to these ends (see Masser 1988a and Rhind and Mounsey 1989 for interpretations of the report's effects). Thus, though the start of the trial phase of the RRLs actually anticipated the formal appearance of the report, the initiative certainly reflects prior discussions between various parties.

THE OBJECTIVES AND FORM OF THE RRLs

The RRL initiative is one of the largest programmes ever launched by the ESRC. Its general objective is to establish regional centres of excellence in the fields of data handling, data base management, spatial analysis, software development, education, training and advice.

The trial phase was set up by inviting applications for prototype RRLs; from the forty or more applications, four organisations were selected to act as RRLs for an initial 18 month period, commencing in February 1987. These initial RRLs covered the South East (Birkbeck College and the London School of Economics), the South West (University of Wales Institute of Science and Technology and the South West Regional Computer Centre, the North (Newcastle and Lancaster Universities) and Scotland (Edinburgh University). As a matter of policy, these were selected in part because of the existence of skilled staff, equipment and software within them. Funding of this pilot or test phase was modest, averaging about \$36,000 per RRL. Each RRL was to use the money to demonstrate real benefits and a demonstrable demand for its skills and services. In practice, all did this by using the ESRC money to leverage larger sums (up to four times the ESRC funds in some cases) from clients, the host institution and elsewhere (Masser 1988b).

The trial phase was evaluated early in 1988 and, between March and July of that year, submissions for main phase funding were evaluated, again from around 40 organisations. All of the original four organisations survived the review but some changes were made in the light of experience : thus Lancaster was set up as a separate RRL and, in SERRL, Birkbeck assumed the lead site role although a significant degree of functional specialisation and site- specific responsibility was introduced. In addition, three other RRLs were selected : the Midlands (based on Leicester and Loughborough universities), Northern Ireland (Queens University, Belfast and the New University of Ulster) and a consortium of two universities and two local governments in Liverpool and Manchester. For this new phase, the total funding was initially \$3.15m over three years but subsequently this has been raised by about an additional \$450k.

The distinctive feature of the British scheme is that - unlike the American one - the regional nature of the organisation was designed in from the outset. Such an approach was not adopted simply to minimise complaints from unsuccessful applicants! It was argued that this took account of existing geographical concentrations of skills and of differences in data collection practice between the four countries making up the UK. Moreover, it took account of two other geographical factors : local variation in research needs and priorities and the virtue in having strong regional, rather than national, training and advisory facilities. The latter point meets a strong plea in the Chorley Report for improved education and training.

Over the three years from October/ November 1988, each RRL is expected to have the following main functions, though the emphasis on each function will vary between the different RRLs :

- (i) Data management. To act as a centre of expertise in the management and integration of data sources, especially at the regional level and below. In addition, the RRL will act as a source of advice regarding available data sets at all levels and maintain linkages with key organisations such as the ESRC Data Archive.
- (ii) Software development. The RRL will obtain and/or produce 'state of the art' software, exploit this in projects and make available documentation, advice and support to collaborating organisations where appropriate.
- (iii) Spatial analysis. Methodological research in the field of GIS and related data handling areas.
- (iv) Education and training. This is intended to cover both research training and professional development. Though primarily orientated towards the research community, it may also cover any other group from which there is a demand.

The model, then, of an idealised RRL is a regional centre of

technical expertise and research excellence which has strong links with its regional community. Such a centre must have the manpower to carry out both basic and applied research and development and to provide advisory and training facilities for its region. It must also have access to suitable hardware and software facilities; whilst three quarters of the RRLs already run ARC/INFO on Vax machines, gifts of equipment and software from vendors (the first from IBM UK) are proving most helpful. Beyond the regional dimension, most RRLs will be expected to achieve national distinctiveness in one or more research areas and act as a national focus for this type of social science research.

It is evident from all this that the ESRC funding alone, on average sufficient for a software person, a spatial analyst and a technical support person at each RRL will be insufficient to meet expectations. This is particularly true since the ESRC funding is strictly limited to a three year term, after which each RRL is intended to be self-sufficient in terms of funding from whatever sources may be available. ESRC's expectation, however, is that their money will again be used to leverage other funds. Assuming a leverage factor of about 2.5, this implies nearly 80 individuals funded to work in this field over the next three years plus all of the research efforts of the academic staff who are paid for out of normal UFC funds. It will be appreciated that the production of tangible products and active collaboration with 'outside' agencies is very much in the interests of both ESRC and the individual RRLs.

A CASE STUDY - THE SOUTH EAST REGIONAL RESEARCH LAB (SERRL)

SERRL's main area of operation covers the traditional South East of England and also East Anglia i.e. an area bounded by a line running roughly from the Wash via Oxford to Bournemouth on the South Coast. Though SERRL staff expect to carry out most of our work in this area (which includes a third of the national population), they also have certain national and even international involvements : for instance, Birkbeck staff are heavily involved in national matters on the next Population Census and on international collaborative work with the French (in relation to the Channel Tunnel and other topics of joint concern) and, though at a very early stage, with the US National Centre for Geographic Information and Analysis. Owing to the expertise of existing staff and accumulated experience, SERRL can claim a national role within the RRLs in regard to both topographic (e.g. Ordnance Survey) and population and planning data and problems.

The lead site in SERRL is the Geography Department at Birkbeck College; this runs Vax mini-computers and workstations, PCs and MacIntosh micros, with access via JANET (the national research computer network) to IBM and ICL mainframes and Cray supercomputers; the graphics equipment includes 'top-end' Tektronix colour terminals bought from the first ever grant given by the University Grants Committee for GIS work, as well as the usual plotters, etc. Software available includes GISs like ARC/INFO, Laserscan software, Apple 'exchangeware', experimental software from universities around the world, teaching packages like MAP2 and numerous

mapping and statistical analysis packages like GIMMS, MAPICS and MINITAB. In the other SERRL site at the London School of Economics, the Geography department has a number of MacIntosh micros and uses MAPICS on the College Vax; it is intended that they will shortly take delivery of PC ARC/INFO.

Arising out of the 'one year, one person' pilot phase, nine SERRL Working Reports were produced and disseminated by the Birkbeck College team and projects with various organisations such as British Rail were carried out. Based on this and related work, papers by Birkbeck staff appeared in five of the first six issues of the International Journal of GIS. A SERRL newsletter was also set up. In parallel with this 'awareness enhancing' activity, pre-existing data bases containing detailed population statistics and infrastructure provision (e.g. the location and type of roads and railways and the London Underground network) in the region were enhanced and linked together.

Basic research work

In many cases, we have little or no quantitative measure of how accurate are the results from the linkage together of geographical data (see, for instance, Rhind 1988); yet such linkage of separately collected data sets is the key to 'adding value' in using a GIS since combinations of the data can be used for purposes additional to those for which the initial data collection occurred. Thus, though only one combination exists of two data sets, no less than 1,048,559 such combinations - not all of which are meaningful - are available from 20 data sets describing 'objects' within the same geographical region. Since data are rarely collected (at least in the UK) on any consistent geographical basis, it is necessary to use the 'space shared' by the geographical objects (polygons such as counties or Health Districts, networks such as streets or streams or points such as geological boreholes or mail delivery points) in each data set to link the data sets together. Such a process frequently involves a process of approximation, especially when the geographical description (e.g. using a unit postcode) is inherently imprecise. Moreover, since geographical data sets are often voluminous and sometimes error-prone, the process is rarely straightforward. SERRL has formulated a modest research programme in the area of data integration problems.

If the results of data linkage are often poorly understood, their display is little better: there is little good evidence on what is an efficient (as compared to attractive) graphic depiction of data - even though maps and diagrams as well as statistical tables are normal output from GIS. Moreover, the techniques of analysis used in GIS are still crude by the standards of those of some human analysts; in particular, human experience and 'soft' or 'fuzzy' information are not readily introduced to the evaluations. Finally, we have at present only very crude ways of conversing with the machine which suggests that future GIS should be able to act on commands given in whatever language or terminology is convenient for the user, whether he is an expert in the transfer of legal titles to houses or an environmental scientist: the development of so-called Natural Languages is a high research priority. Again, SERRL

has a targeted research programme in these areas, part of which is carried out in collaboration with colleagues in other RRLs and other universities (see Rhind, Raper and Green 1989).

The SERRL approach to applied work

SERRL's approach is both pragmatic and eclectic : the method of work varies with the topic. In applied research or development, the preferred method is to work with or for other organisations since this maximises the chances of the work being useful. The operating principle is that all 'applied' work should at least 'break-even' in terms of meeting its costs. 'Profits' are sought wherever possible and all such monies go back into supporting 'core' staff and providing new equipment. In the past, this approach has provided new computers, helped to train staff on secondment (including those on the Birkbeck 'GIS apprenticeships' scheme) and also improved the data base as new information gleaned from projects is added to it. SERRL is based in a university so has a primary commitment to education and research, rather than financial gain: thus frequent use is made (with the agreement of customers) of previous work as case studies in teaching. Wherever possible, publication of such work as scientific papers has occurred, usually together with staff from the customer's organisation.

A central principle is that of independence. Birkbeck forms strategic alliances with carefully chosen partners (and is in the throes of extending and formalising its range of partners) but no relationship may preclude any other strategically important one. Thus Birkbeck runs no less than five systems in-house even though ARC/INFO (for which we were the first site in Europe) is the main 'work-horse' and relationships with ESRI have been extremely close and advantageous over the years. Equally, the Birkbeck geographers now have a relationship with Apple Computers for development of certain GIS teaching materials, as well as using equipment gifted via ESRC by IBM and other equipment purchased from DEC. The SERRL model, then, is of a way of working which normally involves other people and one in which sometimes SERRL leads and sometimes merely contributes, depending on the skills required and available.

Recent or current SERRL work

To illustrate the range and type of SERRL work, we now describe four examples of recent or current Birkbeck projects. The first of these is a study of how satellite remote sensing data from the French SPOT satellite and on-ground data derived from Local Authorities and relating to the London Green Belt, etc can be combined (Barnsley et al 1988). If created as a coherent data base, the accuracy of land use information which can be inferred is greatly improved over those data produced by conventional remote sensing techniques and the range of applications is greatly extended. This project has been carried out in conjunction with planners from the County of Kent.

A second example is the consultancy study now being undertaken by Birkbeck College for the Department of the

Environment (DoE) : this is to define the needs of the Department from the next Population Census and how these could best be met. The Census data are arguably the most important single data source produced by UK government. DoE, for instance, uses it in the calculation of the funds for distribution to local government, in assessments of deprivation across the whole country and in many other research and policy matters. Results are produced in map or tabular form and the census data may need to be linked to other data sets but originating in many sources. In essence, this project involves discussions with all interested parties and the production of costed alternatives; Price Waterhouse, the international management consultants, are acting as sub-contractors to provide certain experience which is lacking in the university domain.

The third example also relates to the next Population Census, to be held in 1991. Thus far, all recent UK censuses have produced statistical tables derived from comparing the answers to different questions and summing the results for standard areas. The degree of cross-tabulation is much greater than in the US census output. As a result, around 4000 values in total were produced for each and every one of the 150,000 different standard areas for which census results are made available after the 1981 Census. Despite this detail, many users can not get the combinations of area and variables they require; on the other hand, much of the standard data is unused. The Birkbeck project, funded by the Census agencies and also by ESRC, is to explore the feasibility of an on-line computer system which would permit users to request (and receive very rapidly) precisely the results they need. The crucial requirement is that no details whatever must be divulged concerning individuals or the households in which they live - hence empirically derived rules which should ensure this constraint is met are being built in and evaluated.

Finally, by way of example, Birkbeck staff have just completed the world's first GIS tutor, called GIST (see Raper and Green 1989). This takes advantage of the Hypercard facilities, superb graphics and ease of use of the MacIntosh computer and permits individuals to explore topics such as data structures, digitising, interpretation of satellite imagery, generalisation and much else. Demonstrations may be selected by the user from the dozens available. GIST contains a searchable bibliography and much generally helpful background information on GIS, plus test questions where appropriate and a log of what the user has tackled and achieved. This will be used in all the 'hands on' portions of our short courses in GIS. To date, four such short courses have been run and all were fully subscribed though only 10% of the participants have been from academic organisations.

The integrated database

One measure of success of all of the basic and the applied research is the complexity, scale and successful use of our database, which is drawn from a multiplicity of sources (maps at different scales, government statistics for large and for small areas, etc). As the centre-piece of its activities, SERRL has the task of building, maintaining and exploiting a

spatially coherent data base of infrastructure and settlement for the whole of our region. This capitalises upon the basic research and new data sets available through project work. At present, the database occupies over 250 Mb. and consists of such features as :

- Settlement, defined by land use as urban areas and by functional importance as urban regions;
- Transport networks, including all roads, (surface and underground) railways, and some utilities;
- Administrative areas, such as Wards, Districts, Counties and Parliamentary Constituencies.
- Planning areas, including Green Belts, Areas of Outstanding Natural Beauty, Development Corporations and Sites of Special Scientific Interest.
- Demographic and household composition data drawn from the Population Census.

CONCLUSIONS

It should be obvious from the above that the RRL initiative differs in a number of ways from the NCGIA initiative, at least as originally designed by NSF :

- (i) the deliberate country - wide spread of researchers in the UK regionally - based model
- (ii) the much heavier emphasis on tangible products, applied work and proselytising in the UK and the open welcome given to collaboration with vendors of software and hardware
- (iii) the relatively short term funding and 'sudden death' end to the UK project, after which self- sufficiency is essential.

That said, there are also many similarities between the two initiatives. The expenditure per annum is very similar. Both initiatives are essentially the products of academics and are guided and largely judged by academics ; hence the judgement of the academic community as well as ESRC on the UK project's success will be strongly influenced by new work reported in well respected journals. In this regard, the British academics might be judged to have an even more difficult task than their American counterparts. Moreover, though the regional scheme demonstrably maximises the numbers of researchers involved and - through inter-RRL competition - generates the maximum level of external support, it provides obvious dangers of duplication in work. Avoiding such duplications when up to 80 researchers are working in 8 centres in 17 institutions is exceedingly difficult and, to this end, common publications, frequent seminars and briefings, etc are planned. Our experience in the pilot phase indicates that only electronic mail makes day-to-day contacts between and even within RRLs a reality.

Finally, though all of the RRLs have been set up as an ESRC initiative, recent developments have prompted joint action by ESRC and the Natural Environment Research Council. NERC has funded a research group in the cartography/remote sensing/GIS area for 20 years (see Rhind 1988b) and today this group of about 12 staff is based in Reading University. It seems likely, however, that a joint ESRC/NERC bid for additional funding of \$2 million specifically for GIS research has recently been agreed by government ; this will be jointly administered by the two research councils which also now head a joint committee of all the research councils on GIS and related topics.

ACKNOWLEDGEMENTS

The authors of this paper are heavily involved in the ESRC's Regional Research Laboratories initiative : IM is the national co-ordinator of it, MB is the technical adviser, JWS leads the South East Regional Research Lab. and DR is a co-director of SERRL. Nonetheless, the views expressed are personal ones and do not necessarily represent those of the ESRC or of any one of the many other individuals involved in the RRL initiative.

REFERENCES

- Abler R. (1987) The National Science Foundation National Centre for Geographic Information and Analysis. Int. Jl. of GIS, 1, 4, 303-26.
- ACA (1988) The Proposed Standard for Digital Cartographic Data. American Cartographer 15, 1, 137pp.
- Andersson S. (1987) The Swedish Land Data Bank Int. Jl. of GIS, 1, 3, 253-64.
- Barnsley M., Shepherd J.W. and Sun Y. (1988) Conversion and evaluation of remotely sensed imagery for town planning purposes. Proc Euro-Carto 7, ITC, Enschede, Netherlands
- Chen Shu-Peng (1987) Geographical data handling and GIS in China. Int. Jl. of GIS 1, 3, 219-28.
- DoE (1987) Handling Geographic Information : the report of the Committee of Enquiry headed by Lord Chorley. Her Majesty's Stationary Office, London.
- ESRC/NSF (1986) Large scale data resources for the social sciences, ESRC, Swindon.
- Kubo S. (1987) The development of geographical information systems in Japan. Int. Jl. of GIS 1, 3, 243-52.
- Masser I. (1988a) The development of GIS in Britain : the Chorley Report in perspective. Environ. and Plan. B 15, 489-94
- Masser I. (1988b) The Regional Research Laboratory initiative. Int. Jl. of GIS 2, 1, 11-22.
- Openshaw S. and Goddard J.B. (1987) Some implications of the

commodification of information and the emerging information economy for applied geographical analysis in the UK. Environment and Planning A, 19, 1423-40.

Ottens H.F.L. (1988) A centre of expertise for geographic information processing in the Netherlands. Paper presented at Euro-Carto 7, ITC, Enschede, Netherlands.

OS (1987) The National Transfer Format, Ordnance Survey, Southampton

Raper J.F. and Green N.P.A. (1989) GIST : an object-oriented approach to a GIS Tutor. Proc. Auto Carto 9

Rhind D.W. (1981) Geographic Information Systems in Britain. in Quantitative Geography ed. N. Wrigley and R.J. Bennett, 17-35, Routledge and Kegan Paul, London.

Rhind D.W. (1987) Recent developments in Geographical Information Systems in the UK. Int. Jl. of GIS 1, 3, 229-41.

Rhind D.W. (1988) A GIS research agenda Int. Jl. of GIS 2, 1, 23-8.

Rhind D.W. (1988b) Personality as a factor in the development of a new discipline : the case of computer-assisted cartography. American Cartographer, 15, 3, 277-89.

Rhind D.W. and Mounsey H.M. (1989) The Chorley Committee and 'Handling of Geographical Information'. Environment and Planning A,

Rhind D.W., Raper J.F. and Green N.P.A. (1989) First UNIX, then UGIX. Proc. Auto Carto 9

Tomlinson R.F. (1987) Current and potential uses of geographical information systems : the North American experience. Int. Jl. of GIS 1, 3, 203-18

THE INSTITUTIONAL CONTEXT OF GIS:
A MODEL FOR DEVELOPMENT

Peter F. Fisher
The Department of Geography
Kent State University
Kent, OH 44242-0001
PFISHER1@KENTVM.BITNET
and

Michael N. DeMers
The Department of Geography
The Ohio State University
Columbus, OH 43210
TS2695@OHSTVMA.BITNET

ABSTRACT

The institutional context in which GIS operate has failed to evolve with either the rate of software and hardware developments, or the numbers of systems installed. This realization has been dawning on the GIS community over a number of years, and has led to studies such as the Wisconsin Land Records Committee and the Minnesota Inventory of Mapping Systems. The failure of institutions to evolve in the context of GIS is rapidly resulting in costly and repetitive efforts in database development and expertise at all levels of government, and in other areas. In this paper we will present a generalized model for the idealized institutional setting of GIS. The model is independent of implementation and could be realized in a continuous spectrum of contexts from fully manual to fully automated, and in size from Federal Government to small city or private company. The main elements are a central database with catalog and coordinating organization, access by as many users as require, secure databases for users if needed, and the implementation of standards.

INTRODUCTION

In most organizations the implementation of Geographic Information Systems (GIS) has been haphazard, and within any one organization any number of subgroups may be operating GIS, of one form or another. This potentially unorganized proliferation has been demonstrated or at least suggested in a number of recent surveys (Craig, 1988; DeMers and Fisher, in prep.; Wisconsin Lands Records Committee, 1987). While the importance of distributed systems and distributed responsibilities has been recognized, they are primarily considered to be a technical issue of database design (e.g. Webster, 1988). Organizational and Institutional issues tend not to have been addressed. This paper discusses a model for the operation of GIS within an organization of any size, and irrespective of either hardware or software.

BACKGROUND

The use of GIS is now widespread among many types of organization in both the public and private sectors. More often than not, this situation has evolved through the progressive implementation of small scale systems within sub-groups (departments) within the organization. Frequently the situation will arise in many organizations when a number of departments are operating GIS, using different hardware and software, but often employing the same databases, or at least having data types in common. These data types have usually been digitized in common.

A CASE STUDY

In a recent study of GIS within state government in Ohio, DeMers and Fisher (in preparation) found four systems operated by the Ohio Department of Natural Resources (ODNR), the Ohio Environmental Protection Agency (OEPA), the Public Utilities Commission of Ohio (PUCO), and the Ohio Department of Transportation (ODOT). The first two of these are primarily concerned with natural resource analysis, while the latter two have been developed for facilities management. The four systems all use different software, hardware and databases. Nothing is held in common, although there has been some communication among the groups, and occasional transfers of data have been conducted. Within the databases of all four organizations, however, political divisions, the drainage network and some form of the road network have been digitized separately (although to different levels of resolution). Several other data types occur in more than one of the four systems.

The situation in Ohio is not untypical, and even within that state's governmental organization it is known that a number of other departments are currently exploring the potential of GIS. Systems are also being implemented in a number of cities within the state, including both the largest, namely Cleveland, Columbus and Cincinnati, and smaller ones such as Akron, Kent, Medina, and so on. At present, there is no framework, organization or even individual through which existing or potential systems may interact. The potential for repeated collection of the same spatial data is enormous, and since the data collection phase is undoubtedly the most costly in establishing most GIS, redundancy in this phase may represent a considerable drain on the public purse.

A number of factors must be considered in suggesting any steps towards reducing the cost of digitization by data sharing. First, within the organizations reviewed, there was considerable concern for data security and confidentiality. Thus OEPA considered that they held highly sensitive data which would not be appropriate for

public examination, and PUCO believed that several utilities would be unwilling to share data if it had not been confidential. Second, each GIS operation considered that it was fulfilling a function of considerable utility within its own department. Finally, it is necessary to recall that the different departments have very different roles, and the different systems are used for different functions. Thus ODOT and PUCO are involved in facilities management, while OEPA and ODNR are natural resource management systems. Furthermore, ODOT has regulatory control over the state road network, OEPA regulates and monitors environmental impacts, and PUCO is concerned to monitor the activities of utility companies. ODNR, on the other hand is involved in inventory of natural resources and advising on their exploitation.

A MODEL FOR INSTITUTIONAL REVISION

Within Ohio state government, and in many other organizations, GIS implementations are at present operating in isolation, without any official form of contacts among the different groups. To prevent redundancy in data collection and in expertise institutional revision is required. Figure 1 presents a model of a form of interaction which involves a number of features intended to be sensitive to the requirements of different departments, as well as to the need to share data and expertise. The model is, however, independent of both software and hardware used by the different GIS operations, and may be implemented in a number of ways.

The model has a number of features:

1. A central base of statewide spatial data. All data types are available to all users. The Central Database itself has a number of parts:
 - a) Central Catalog where information is held on all data types made available by the participating organizations;
 - b) Collections of data for which responsibility is clearly targeted to a particular organization.
2. Some number of operating GIS (four exist in Ohio at present) each of which have:
 - a) a secure database of information that is considered confidential for that GIS operation.
3. Rigorous implementation of standards to facilitate data transfer (American Cartographer, 1987).

The model can be implemented at almost any level of automation and of standardization of hardware and software:

- 1) At the most automated, the system could involve a number of users of a central GIS, all users operating on the same computer system, with local workstations, and using a single software package. In some respects this is the most logical form of implementation.
- 2) At one level down, the participating GIS operations could each operate stand alone GIS each with different software and hardware, but connected by a network. The catalog would be maintained by a central organization, but the data could either be held by that organization, or by the individual GIS operations, simply marking data as secure or open.
- 3) The least automated version of the system would have no actual central database, and no network of interaction. Data types that operations are prepared to share would simply be transferred manually on tape or disk form the originating GIS to a user.

In short, the catalog of data digitized and the details of that data contained in the catalog (as specified by standards) are central to the model. Maintenance of that catalog will allow users to identify when particular data for an area has been previously digitized and at what resolution digitizing took place, and so establish whether the data may be useful to them, in place of redigitizing the information.

CONCLUSION

A model of how a network of GIS operations might be arranged has been proposed. The arrangement is particularly sensitive to two concerns expressed here: First, that where desirable data security should be assured, and second that the risk of data redundancy within the organization should be minimized. Further, the model can accommodate any number of different software packages, and can be implemented at a number of levels of automation.

ACKNOWLEDGEMENT

We wish to thank all those who assisted us in the review partially reported here, namely Tom Beard, Wayne Channell, David Crecelius, Charles Groves, John Helms, Gail Hesse, Ava Hottman, Gene Johnson, and Terry Wells of the various Ohio governmental organizations.

REFERENCES

American Cartographer, 1987, The Proposed Standard for Digital Cartographic Data: American Cartographer 15: 1-142

Craig, W.J., 1988, Minnesota Inventory of Computer Mapping: Software, Applications, Graphic Data Files and Expertise, Regional Mapping Consortium, St Paul, Minnesota

DeMers, M.N. and Fisher, P.F., in preparation, Evolution of Statewide Geographic Information Systems in Ohio: A Case Study of the Institutional Context: Submitted to International Journal of Geographical Information Systems

Webster, C., 1988, Disaggregated GIS Architecture: Lessons from recent developments in multi-site database management systems: International Journal of Geographical Information Systems 2: 67-79

Wisconsin Land Records Committee, 1987, Modernizing Wisconsin's Land Records, Final Report of the Wisconsin Land Records Committee, Madison, Wisconsin

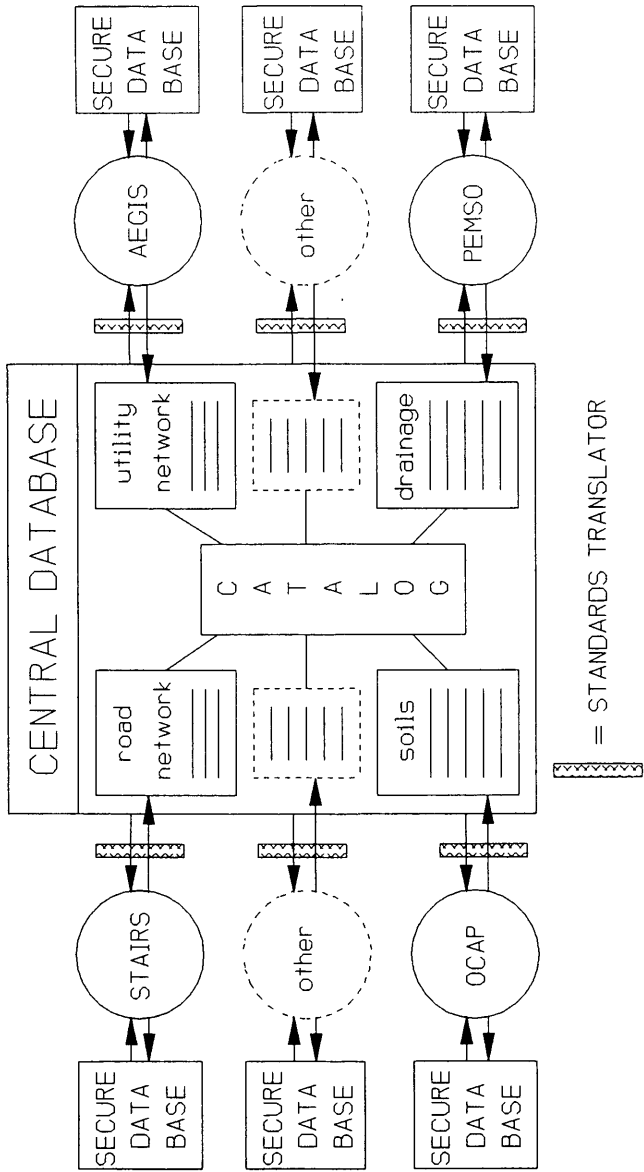


Figure 1

Diagrammatic representation of the model for development within organizations using a number of GIS installations.

THE POWER OF SYMBOLOGY IN THE GIS WORLD

Michael E. Gentles
Senior Consultant
Synercom Technology Inc.
Houston, Texas 77042

ABSTRACT

This paper will examine and explain some unique ways for analyzing data with the use of symbols. Symbolic maps are an effective way to display spatial information. Symbols can show a number of variables with the same display. Sixty to seventy percent of the data within an organization is geographic related and has a geographic identifier. Types of data that can be displayed using creative symbology are land-information/parcel data, socio-economic data, load studies using meters and transformers, and incident reports for crime, such as the locations where cars were stolen and recovered and violent crime has occurred.

When analyzing data with a geographic spatial placement and location, the power of the symbol is much greater than the standard 2-dimensional or 3-dimensional perspectives of thematic analysis. This paper will explain the uses of symbol shape, size, and color to assist in the analysis of data.

INTRODUCTION

The use of symbology with automated mapping and facilities management in the areas of data analysis, data validation, and quality control provides an excellent and powerful tool in which to communicate to the map reader additional underlying phenomena about a spatial feature. Maps can be very simple showing a two dimensional difference or very complex, showing many variables within a symbol. As the maps become more complex they must be able to show the quantitative geographical phenomena as spatial attributes.

The examples and concepts used throughout this paper will concentrate on maps that range from a scale of 1:500 to 1:5000. The examples and data bases will come from municipalities using multipurpose cadastre, public works departments, and utility companies with whom the author has worked with in the past.

BACKGROUND

With the rapid expansion of computerized mapping using attribute information attached to the graphic symbology, the means for improved data analysis and data validation can be accomplished with the use of symbology. Large tabular listings used in the past for data analysis and data validation

now can be replaced or supplemented by multi-symbolized maps to assist the analyst or quality control personnel in checking and analyzing data.

But where does this attribute data come from? The attribute data attached to the graphic symbology comes from a number of different sources, including existing hardcopy, manually drawn maps, or bulk loaded from computerized files. Once this attribute data is attached to the symbology there are still some very important questions that need to be answered. How valid was that data from the original sources? What checks were used to analyze the quality of the data prior to being attached to the data base? Is the data spatially correct?

Through the use of computerized mapping systems, the user has a choice of displaying the data in a number of ways when boolean criteria is applied against the attribute data.

SYMBOLOLOGY

Symbology on maps are displayed in three classes;

Point	Representing a single geographic location, such as the visual centroid of a lot, a location of a device such as a electrical transformer, or water valve
Line	Representing a linear feature such as an electrical primary conductor, a water line, or a shoreline
Area	Representing a geographic area (polygon), such as a census tract, the area of a lot, or other thematic presentation

There are two different types of symbology, replicative and abstract. Replicative symbols are those that are designed to look like their real world counterparts; they are only used to represent tangible objects. Abstract symbols generally take the form of geometric shapes such as circles, squares, and triangles. (2 p.20) Coast lines, trees, houses, and cars are examples. Base map symbols are replicative in nature, whereas thematic-overlay symbols may be either replicative or abstract.

The visual representation of a symbol when built against the attribute data, can tell the user many different things about the symbol. This is done by using the shape, the size, the color or any combination of these representation. The shape of a symbol has already been addressed. The size of symbol can be varied as to a given set of ranges. The lower values of a range would be shown with a smaller sized symbol, while a larger sized symbol would indicate a larger value. Color can easily be used to represent different features or be used to show different ranges.

EXAMPLES

A symbol within a lot or parcel of land can represent a number of different things when the symbol is determined by one or more of the associated attributes of that parcel centroid. Table I. illustrates how the number of bedrooms for a parcel could be shown.

TABLE I

Number of bedrooms	Shape	Color	Size
2 or fewer	plus	green	one half normal size
3	triangle	blue	normal size
4	square	red	one and a half times normal size
5 or more	star	purple	twice normal size

Using either the shape, the color, or the size of a symbol, a user analyzing a map portraying the number of bedrooms would easily be able to distinguish by the symbology which parcels had two bedrooms, three bedrooms, four bedrooms or five or more bedrooms. When spatially analyzing a large geographic area an anomaly will stand out. If a neighborhood had homes with only two and three bedroom homes, a five-bedroom symbol could warrant additional research as to the validity of the data. There may be other underlying data that would validate the number correct.

In Table II, parcel number four, indicates eight bedrooms, but further examination of that record reveals the parcel was zoned for a four plex, indicating eight bedrooms is feasible. If parcel number four had been zoned as an R-1 single-family residence, then the number of bedrooms could require verification.

TABLE II

Parcel Number	Zoning Code	Land Value	Improvement Value	Square Feet	# of Bedrooms
1	R-1	\$22,000	\$45,000	1800	3
2	R-1	\$19,800	\$48,000	1467	2
3	R-1	\$10,001	\$55,000	1782	4
4	R-4	\$35,000	\$75,000	2412	8
5	R-1	\$21,000	\$42,000	2162	3
6	R-1	\$29,900	\$44,000	1944	3
7	R-1	\$22,000	\$51,000	1827	4
8	C	\$60,100	\$83,000	3775	-

When analyzing the maps, more than one dimension is needed to analyze the underlying reasons for an apparent anomaly. As in the above example, if zoning was to be shown by one method, e.g. color, number of bedrooms by the shape of the symbol, and the size of the structure by the scaled size of the symbol, it would be simple to determine if the data appeared correctly.

With the use of computer-generated maps, a single symbol within a lot or parcel could portray a number of different variables, such as:

- range of improvement value
- the square footage of the residence
- the number of bedrooms

The range of improvement value could be shown by using different colors to indicate the various ranges. A green symbol could represent an improvement value less than \$40,000.00, a cyan-colored symbol would represent a range of \$40,000.00 to \$50,000.00, a magenta-colored symbol would represent an improvement value of \$50,000.00 to \$60,000.00, and a red-colored symbol would indicate that the improvement value was greater than \$60,000.00. The square footage of the residence would be shown by the scaled size of the symbol, a smaller size symbol indicates a low square footage, while a bigger size symbol indicates a larger square footage residence. The number of bedrooms would be shown by the shape of the symbol. Two bedrooms and less would be indicated by a circle, a three-bedroom residence would be shown with a triangle, a four-bedroom residence with a square, and a residences with five or more bedrooms would be shown with a star. This same symbol symbology could be made even more complex by combining a second symbol on top of the first using different colors, different scales, and different shapes to add three other variables to the one symbol location.

Using the choropleth/thematic mapping method to indicate land values will not show any large discrepancies. As an example, creating a land value choropleth map (Figure 1), from the attribute data from Table 2, and using the ranges in Figure 1, \$0.00 to \$9999.99 is shown in the lightest texture, thru \$50,000.00 and above indicated by the darkest texture. Two adjacent parcels appearing to be the same size, show in two different textures. The difference in the land value between these two parcels could be \$1.00 apart or \$19,999.00 apart. The reviewer or users need some way to evaluate the validity of the data. If the users had used symbology with the size of the symbol to indicate the land value and the symbol shape to indicate zoning, as in Figure 2, any large discrepancies could be easily distinguishable and possible conflicts resolved.

Figure 1

CHOROPLETH MAP

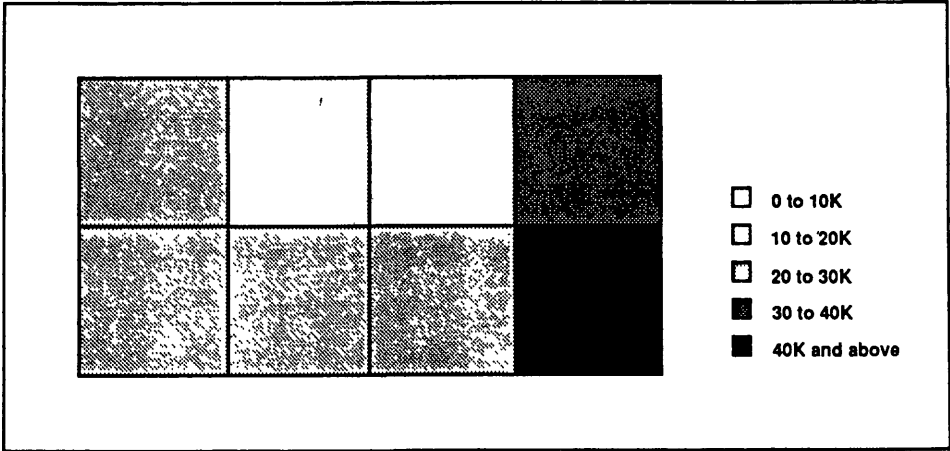
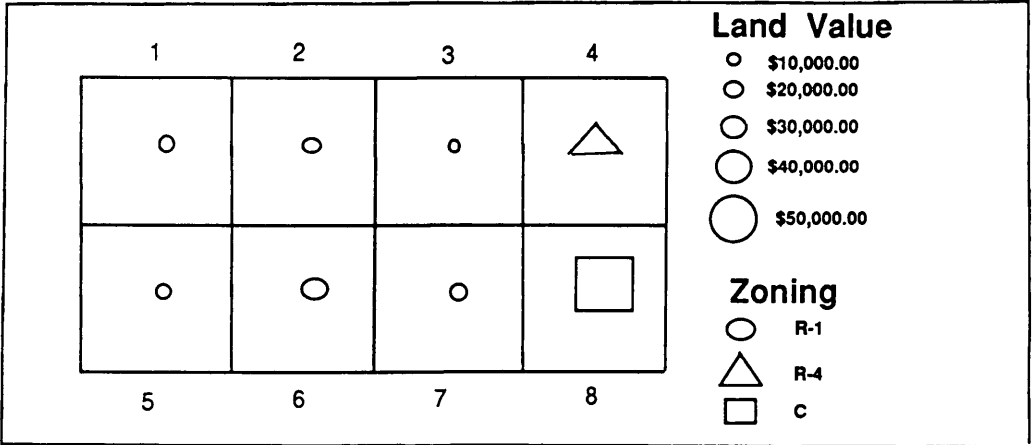


Figure 2



Another example of the use of symbols for data validation is with the use of linear maps. Linear maps can also have complex symbology to indicate two or three different types of symbology. Electrical distribution will be used in the following example and will include color for the electrical phase or phases, the thickness of the line will indicate the KVA of the line and the symbology within the line will indicate the material of the line.

In working with electrical distribution networks, there are three phases for the primary conductor, A, B, and C. Using the three primary colors, red, yellow, and blue one can create line colors that will visually point out any phase errors very quickly. If all A phase lines are done in red, all B phase lines are done in Yellow, and all C phase lines are done in Blue, then the following colors are created

- ABC = white
- AB = orange
- AC = violet
- BC = green
- A = red
- B = yellow
- C = blue

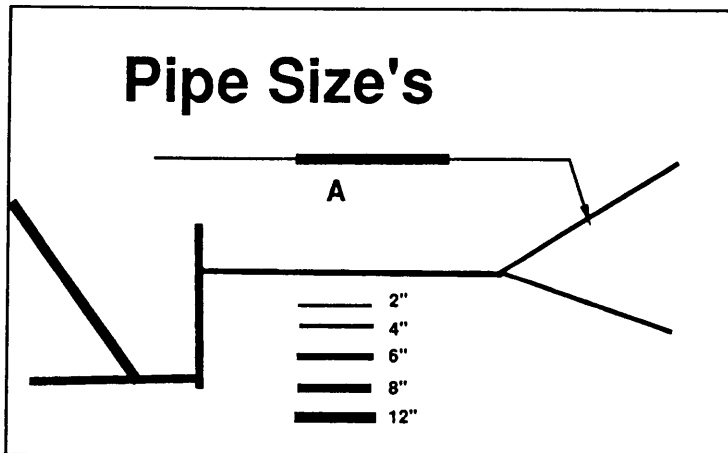
With this color scheme set up, the user can now visually see that the phase of the electric lines and the associated electrical devices have the proper phase conductivity by working with the following rules;

- A white line can have any combination of device colors attached to the line or branching from the line.
- An orange line is fed by a white line and can only have a red, orange, or yellow device attached or branching from the line.
- A violet line is fed by a white line and can only have a red, blue, or violet device attached or branching from the line.
- A green line is fed by a white line and can only have a blue, green, or yellow device attached or branching from the line.
- A red line can only be fed by a white, orange, or violet line and can only have a red device attached or branching from the line.
- A yellow line can only be fed by a white, orange, or green line and can only have a green device attached or branching from the line.
- A blue line can only be fed by a white, green, or violet line and can only have a blue device attached or branching from the line.

The type of primary KVA line can be symbolized by a line pattern indicating whether it is a low, medium or high-rated KVA type of primary. A 2-KVA line would be indicated by the line being one pen width wide. The 7-KVA lines would be portrayed as two pen widths wide and the 14-KVA primary lines would be portrayed as lines being three pen widths wide. At the locations where the line width changes, indicating a change in the KVA, there must be a step down/up device at the location. If there is not a step down/up device at this location there is an apparent error either in the data or in the conversion effort.

Another example of using linear maps to check out the data attribute of size for water and/or gas lines is to plot each different size pipe in a different color or line width. By plotting the different size water lines with different shading, line widths, or colors, sizes that appear to be incorrect such as an 8-inch water line connected on each end to a 2-inch line would stand out. This is indicated in Figure 3 next to the 'A', and that piece of 8-inch pipe would be marked for further investigation. As mentioned above, with the electrical devices being plotted in different colors to ensure correct phasing, all devices attached to the pipe lines such as valves, regulators etc. would have the size of the device plotted in the same color to help ensure the validity of the data attached to the pipe lines.

Figure 3



There are limitless other unique types of plots using different types of symbology that can be produced. Recently I had a chance to review some crime incident plots produced for a municipality that indicated the type of crime. What was unique about these plots was the symbology used. Armed robbery was shown with a symbol the shape of a gun, a stabbing was shown with a knife, stolen cars symbol was a car. To assist in the analysis of these crime, the time of day could have been indicated by the color, letting the watch commander know if the crime was being committed on his shift.

Another municipality uses the size of the symbols to indicate water pressure for all of the fire hydrants. This allows the fire department when responding to a call to use pressure as well as proximity to the fire, to determine which hydrant to use for the fire.

A municipality on the west coast has all the Class A petroleum pipelines, those pipelines that are highly flammable, that pass through the city displayed in red. The materials carried through the line is plotted by using different line patterns.

CONCLUSION

What we have attempted here is to indicate how symbology can be used for data analysis and data validation. The options in working with multiple symbology using shape, size, and color, is almost limitless and the author encourages the readers to experiment with their own data in producing different types of maps.

REFERENCES

- (1.) Auto Carto IV Vol II p158 "Interactive Computer Mapping Applications to Criminal Justice Planning in Three Virginia Cities"
Leonard J. Simutis, Assistant Dean
Todd A. Scott, Instructor
Division of Environmental and Urban Services
Virginia Tech, Blacksburg Virginia 24061
- (2.) Principles of Thematic Map Design. Borden D. Dent
Copyright 1985 by Addison Wesley Publishing Company, Inc.

ON THE DESIGN OF GEOGRAPHIC INFORMATION SYSTEM PROCEDURES

By

J. Armando Guevara, Ph.D.
Software Engineer

Environmental Systems Research Institute, Inc.
380 New York St.
Redlands, California 92373

Telephone: (714) 793-2853 Ext. 208

AUTO/CARTO 9
April 2-7, 1989 Baltimore, MD

ABSTRACT

This paper identifies the building blocks that have played a major role in the design and implementation of current geographic information system procedures. It then examines and proposes the following six continuity concepts as unifying elements of an evolutionary GIS:

1. Functional continuity: the ability for a GIS to have a transparent functional flow of control.
2. Data base continuity: the ability of a GIS to manage giant amounts of data on a distributed system as one logical data base and have multi-user access.
3. Data structure continuity: the coexistence of vector, lattice, and raster data structures under one data model.
4. Knowledge continuity: the utilization of artificial intelligence techniques to create data base model usage schemas and create application procedures.
5. Human interface continuity: what makes a good GIS interface.
6. Data transfer continuity: the ability of a GIS to exist and transfer data independent of the hardware platform.

INTRODUCTION

A geographic information system (GIS) is composed of a set of building blocks termed geographic information system procedures. A geographic information system procedure is an abstract algorithmic function of a GIS that allows one to select, process, and update elements from a spatial data structure (SDS) and/or spatial data base (Guevara, 1983). Based on this, a GIS can be defined as a model composed of a set of objects (the spatial data structures) and a set of operations (GISP) that perform transformations and/or queries on the spatial objects. Unlike any other information system, GIS has the particular characteristic that its operations are mainly of spatial nature, thus a GIS is part of a major group of systems called Spatial Information Systems (SIS). The elements modeled by a SIS are generally imbedded in two-dimensional space and in some instances in two-and-a-half and three-dimensional space. In addition to the elements it manipulates being spatially located, the elements themselves possess a set of attributes that can be qualitatively or quantitatively defined. These attributes can not only give a description of the spatial elements, but can also become time components (changes in time of the spatial data base). Given this particular nature, GISP must be able to both query/transform the spatial elements and the attributes associated with those elements.

GISP are categorized to be the primitive operators of a GIS. In this sense GISP can be:

- | | |
|-----------------|----------------------------------|
| a) SELECTORS | - Capture, select spatial data. |
| b) RECOGNIZERS | - Structure/search spatial data. |
| c) PROCESSORS | - Process spatial data. |
| d) TRANSFORMERS | - Output spatial data. |

This paper identifies the major building blocks that have played a key role in the design and implementation of GISP:

1. The use of geometry as the mechanism to digitally model the location of spatial elements.
2. The study of computational geometry; a better understanding of the digital representation of geometric algorithms and their numerical aberrations.
3. The use of topology to digitally model the relationship among data elements.
4. The local processing concept: the managing of large amounts of spatial data under limited RAM.
5. The fuzzy intersection concept: used to make the polygon overlay problem tractable at the implementation level.
6. The geometric simplification concept: used to simplify the geometric complexity of GIS objects.
7. Data base management systems: the relational model.

In retrospect, these concepts have served their purpose and are now the cornerstones of many implemented GIS. What is required now is to evaluate what is needed next in this growing and demanding technology. This paper then examines and proposes the following six continuity concepts as elements of a unified and evolutionary GIS:

1. Functional continuity: the ability for a GIS to have a transparent functional flow of control.
2. Data base continuity: the ability of a GIS to manage giant amounts of data on a distributed system as one logical data base and have multi-user access.
3. Data structure continuity: the coexistence of vector, lattice, and raster data structures under one data model.
4. Knowledge continuity: the utilization of artificial intelligence techniques to create data base model usage schemas and create applications procedures.
5. Human interface continuity: what makes a good GIS interface.
6. Data transfer continuity: the ability of a GIS to exist and transfer data independent of the hardware platform.

BACKGROUND HISTORY

It has been almost eight years since the first introduction of the ARC/INFO system. This system introduced for the first time a widely distributed and used operational GIS. Conceptual aspects of GIS have been around now for over 25 years. The gap between theory and practice began to be broken in the mid 70's and the technology has really taken off in the latter part of the 80's. If we classify the major breakthroughs that created this bridge we have:

- a. The formal study of geometric algorithms via computational geometry (the study of spatial searching schemas, the study of spatial data structures) (Shamos et. al. 1976).
- b. The use of topology to establish the spatial context for the geometric elements being digitally represented.

- c. The organization of the spatial elements in a Data Base Management System.
- d. The advent of high-power-low-cost computers.

FUNCTIONAL COMPONENTS OF GIS DESIGN

The most important concepts introduced in the functional design of a GIS have been that of the Local Processing Concept (Chrisman, 1976) and Fuzzy Overlay (White, 1978; Guevara, 1983). The Local Processing Concept made tractable at the implementation level the processing of large amounts of data given a limited amount of RAM by using the spatial properties of the data of location and orientation. The Fuzzy Overlay concept allowed for the implementation of the most powerful function of a GIS, that of data integration.

As knowledge was gained on the behavior of spatial algorithms, a functional categorization emerged that did away with functional continuity. Functional continuity refers to the ability in a GIS to be able to access any data set (or portion thereof in a seamless data base) and apply operators without the system losing track of the data environment and history of the operations performed. Functional continuity would allow access to all GISP within one process environment. Although this has a tremendous power, it is not without its user interface complications.

The functional categorization introduced gave way to the basic architecture of a GIS: data input, data base management, analysis, and output. Within each category, different means have been created to handle the user interface. Menu and command driven functions have become the main ways of interaction. The functions have a proper protocol to internally deal with the processes. Some are action driven while others are environment driven.

Action driven functions produce an immediate feedback to the user (e.g., draw a map). Environment driven functions have a cumulative effect that ends in an action driven function. Action driven functions are easy to explain and use. Environment driven functions pose a variety of user interface problems.

A functionally continuous GIS would be mostly environment driven. Such a system would require of knowledge environment and function tracking procedures. Recognizing the importance of user interaction with a GIS is the concern of the Human Interface Continuity Principle and is key to the life appreciation or depreciation of the system (how easy or complicated it is to learn and use).

DATA MODEL COMPONENTS OF GIS DESIGN

The most important concepts introduced in GIS that allowed one to digitally model spatial relations was that of topology (Corbett, 1975) and the relational data base model (Codd, 1970). It is interesting to find that it really has not been until recently that the power of this notion has been widely accepted at the implementation level of GIS. ODYSSEY was the first system to implement it (Dutton, 1978), then ARC/INFO (Aronson et al. 1983).

The various data structures introduced to handle geographic data (Morton sequences, Peano curves, quad trees, R-trees, B-trees, etc.) and their general definition and/or implementation (vector, raster, lattice) were to guide the definition of the GIS data model in the sense of only dealing with one particular data structure at a time. To achieve a continuous data model in the true perspective of not

just spatial continuity but data integration also, the design of a GIS system must take into account the integration and management of all these data types (data structures). This would allow a GIS system to handle planimetric data, surface data, and imagery data.

In the early 70's the relational data model was introduced along with mechanisms to express relations between stored items (a relational algebra). Because the ubiquitous geographic matrix (rows, location, columns, description) fitted so naturally the relational model, this made the transition quite natural and simple to implement. A one-to-one relationship between the geometry and the descriptive data could be implicitly accomplished.

SPATIAL DATA MODELS

The ultimate task of a GIS is to model some aspect of a spatial reality. The model should include enough information that would allow its user to obtain answers to queries and infer situations that otherwise would not be possible. We can identify two types of models:

- a. a generic functional model
- b. a specific derived model.

A generic functional model (GFM) is a model made of basic spatial primitives: points, lines, and areas. The model holds descriptive data about the primitives, but does not know about existing relationships. The model is functionally driven (i.e., any further inference about the data aside from primitive location and basic description is obtained via spatial operators). The GFM is an open model that requires only very basic knowledge about the spatial primitives being stored.

A specific derived model (SDM) is a model derived from established relationships among the spatial primitives, and a linkage is created among compounded spatial primitives and their descriptive data. The SDM requires a well-understood knowledge of how the GIS is going to be used and what it is going to model.

The relational approach to spatial data handling falls under the GFM category, while the object-oriented approach falls under the SDM. It is important to understand that these two models are not mutually exclusive (i.e., a GFM can be used to support a SDM). However, the absence of an underlying GFM in a SDM raises flexibility and performance issues.

The GFM has the following characteristics:

- a. It should allow for dynamic relationship construction via spatial operators.
- b. Compounding of spatial primitives should be done efficiently without restrictions or constraints. The compounding would still yield a (more complex) GFM.

Internally, the GFM follows a similar structure to that described in Guptill (1986) with the exception that the lowest level of the model relationships are not explicitly stored.

The SDM has the following characteristics:

- a. Relationships between spatial primitives are pre-established in the model based on behavioral, procedural, and transactional facts. These facts make the SDM schema.
- b. Mutations on the spatial primitives should be done efficiently. Mutations such as aggregation (compounding), disaggregation (uncompounding) would still yield a (more complex or simpler) SDM.

The SDM would be the basic model for object-oriented transactions as those described in Kjerne (1986) and fundamented in Cox (1986) and Bertrand (1988).

The GFM and SDM should allow for the following types of data base queries:

- a. Spatial Context: Given an unambiguous geometric definition, extract from the data base all elements selected by the geometric definition.
- b. Spatial Conditional Context: Given an unambiguous geometric definition and a condition expressed in terms of the stored descriptive data, extract from the data base all elements selected by the geometric definition and that suffice the descriptive condition given.
- c. Descriptive Context: Given a descriptive data element, extract from the data base all elements that match the one given.
- d. Descriptive Conditional Context: Given a descriptive data element and a condition expressed in terms of the given element, extract from the data base all elements selected that suffice the descriptive condition given.

The conjunction of a GFM and a SDM would give the user the ability to perform spatial operations at various levels of complexity and integration. GFM and SDM bring the ability for a GIS to be flexible and schema independent.

Finally, both the GFM and the SDM should maintain the data base continuity concept (i.e., preserve the notion of a continuous physical space underlying the data model).

TOWARD AN ADAPTABLE SPATIAL PROCESSING ARCHITECTURE

A modern GIS is expected to be able to integrate a different variety of data sources; these data sources will be used in many ways and also under a wide range of support decision making. The nature of separate user views of the same data base accompanies a series of (sometimes conflicting) demands to the GIS designer that must somehow be met to guarantee the usefulness and longevity of the system. In synthesis, a GIS is a multidisciplinary tool that must allow for interdisciplinary support. Specialized spatial information systems are not multidisciplinary tools, thus are very restrictive in regards to what can be done with them.

An Adaptable Spatial Processing Architecture (ASPA) is what is needed to meet the demands of both multidisciplinary and specialized applications. ASPA fundamentals are based on a GFM that has a set of functional (GISP) primitives clearly defined that allow the automatic construction of a SDM. ASPA has to be designed based on

the six continuity criterions given above. In this respect, ASPA would be an expert monitor based on a high level language consisting of spatial operators that have definable hierarchical constructs. These spatial operators can be organized following a programmable schema that would allow to generate the SDM. ASPA would work in conjunction with a data base management system (DBMS). The DBMS would respond to both spatial and non-spatial operators. The heart of ASPA and the DBMS would be GFM.

The spatial operators and spatial data structures that ASPA is built upon are based on the five basic software engineering principles of modularity, encapsulation, localization, uniformity, and confirmability (Jensen et al., 1979) applied through the concept of abstraction at the design level of the SDM (Guevara, 1981).

Levels of Abstraction

Levels of abstraction were first defined by Dijkstra (1969). They provide a conceptual framework for achieving a clear and logical design for a system. The entire system is conceived as a hierarchy of levels, the lowest levels being the most specific. Each level supports an important abstraction.

Each level of abstraction is composed of a group of related functions. One or more of these functions may be used by functions belonging to other levels; these are the external functions. There may also be internal functions which are used only within the level to perform certain tasks common to all work being performed by the level and which cannot be referenced from other levels of abstraction.

Levels of abstraction, which will constitute the partitions of the system, subsystem or procedure, are accompanied by rules governing the interrelations between them. There are two important rules governing levels of abstraction. The first concerns resources: each level has resources which it owns exclusively and which other levels are not permitted to access. The second involves the hierarchy: lower levels are not aware of the existence of higher levels and therefore may not refer to them in any way. Higher levels may appeal to the external functions of lower levels to perform tasks and also appeal to them to obtain information contained in the resources of the lower levels (Liskov 1972).

The abstraction of a procedure begins at the level of specification and the details that clarify the abstraction are added at the implementation level (Parnas 1972).

In this respect, the lowest level of abstraction is composed of the GFM, a clearly defined set of spatial operators (selectors, processors, recognizers, transformers) and a DBMS. The building blocks of the SDM are then those based on ASPA.

Data and System Independence

A GIS must be data and system independent. Multiple functional mappings should be allowed between the GFM, SDM, and any external data transfer operator. Similarly, the levels of abstractions induced in the GIS should allow the GIS to perform identically on different computers with no user intervention when doing the functional mappings.

CONCLUSION

GIS technology has finally taken off. However, as users become more sophisticated and demanding, we begin to discover how good a GIS has been designed. The notions of continuity presented above are the most important issues that need to be covered for a successful design. In my experience, along with the internal algorithmic robustness and data base consistency and integrity, flexibility and user friendliness (magic words) are today the most relevant points to be considered from the outcome of the design.

We should avoid making the mistake made during the 70's where authors entrenched themselves about whether raster data structures were better than vector structures. None and both were the answer. As we uncover the usefulness of concepts such as objects (object data bases, object oriented software engineering), we must not lose track of the flexibility geographic information systems must have. GIS are multidisciplinary tools. Fixed schemas will hinder GIS usage.

A solution has been explored here, whereby a GIS based on the six continuity principles given is able to support a Generic Functional Model (basic primitives, tool kit) that can generate via an Adaptable Spatial Processing Architecture, a Specific Derived Model (features, objects).

ACKNOWLEDGEMENTS

This work was possible thanks to support from project 9845/190 of the Environmental Systems Research Institute. This paper is part of ESRI's ongoing research program on GIS technology.

REFERENCES

- Aronson, Peter and Morehouse, Scott (1983), "The ARC/INFO Map Library: A Design for a Digital Geographic Data Base," in Proceedings of AUTO-CARTO VI, Ottawa, Canada.
- Bertrand Meyer (1988), Object-Oriented Software Construction, Prentice Hall International.
- Codd, E.F. (1970), "A Relational Model of Data for Large Shared Data Banks," CACM 13, No. 6.
- Chrisman, Nicholas (1976), "Local vs. Global: the scope of memory required for geographic information process," Internal Report 76-14, Laboratory of Computer Graphics and Spatial Analysis, Harvard University.
- Corbett, J.P. (1975), "Topological Principles in Cartography," Proc. AUTO/CART 2, Reston, VA.
- Dutton, Geoffrey (1978), "Navigating in Odyssey," in Harvard Papers on GIS, Vol. 6, Harvard University.
- Cox, B.J. (1986), Object Oriented Programming, An Evolutionary Approach, Addison-Wesley.
- Dijkstra, E.W. (1969), Structured Programming, Prentice Hall.
- Guevara, J. Armando (1983), A Framework of the Analysis of Geographic Information System procedures: The Polygon Overlay Problem, Computational Complexity and Polyline Intersection, Unpublished Ph.D. dissertation, Geographic Information Systems Laboratory, State University of New York at Buffalo.
- Guevara, J. Armando (1981), "Cartographic Data Structures: Abstraction," Unpublished paper, Geographic Information Systems Laboratory, State University of New York at Buffalo.
- Guptill, S.C. (1986), "A New Design for the U.S. Geological Survey's National Digital Cartographic Data Base", Proceedings, Auto/Carto London, Vol. 2, 10-18.
- Jensen, R. and Tonies, C. (1979), Software Engineering, Prentice Hall.
- Kjerne, D. (1986), "Modeling Location for Cadastral Maps Using an Object-Oriented Computer Language," Proceedings, URISA, Vol. 1, 174-189.
- Liskov, B. (1972), "A Design Methodology for Reliable Software Systems," Proceedings, Fall Joint IEEE Computer Conference.
- Morton, G. (1966), "A Computer-Oriented Geodetic Data Base, and a New Technique in File Sequencing", report prepared for IBM Canada Ltd., Toronto.
- Parnas, D.L. (1972), "A Technique for Software Module Specification with Examples," Comm. ACM, Vol. 15, No. 5.
- Shamos, M.I. and Hoey Dan (1976), "Geometric Intersection Problems." 17th Annual Symposium on Foundations of Computer Science, pp. 208-215.
- White, D. (1978), "A Design for polygon overlay", in Harvard Papers On Geographic Information Systems, Vol. 6, Harvard University.

Performance Testing of Gridcell-Based GIS

Sherry E. Amundson
University of Hawaii at Hilo
Hilo, HI 96720

ABSTRACT

The advent of the non-commercial microcomputer-based gridcell GIS brings a host of first-time users who need to know what to expect in terms of processing time. This implies a need for more comprehensive performance evaluation than has been done on GIS in the past. The present study demonstrates the use of the formal performance evaluation methodology in the micro-computer-based setting. It measures the performance of four GIS functions of the OSU MAP-for-the-PC software under a number of varying workload and operating conditions. Commercial performance profiling software is used to monitor program performance internally.

INTRODUCTION

Several non-commercial PC-based GIS, initially introduced as teaching tools or systems for small government projects, have recently been made available at a modest price. Almost all of them organize data using a gridcell structure.

A thorough evaluation of the performance of these systems -- in terms of processing time and disk utilization -- is needed by a potentially large set of first-time users with a potentially diverse set of requirements. Users need to know what to expect. (Should they wait at the terminal for their results? Should they go out for coffee and come back later? Should they let the program run overnight?)

Processing time varies with the size and complexity of the input data, and with variations in the computing environment. Therefore an assessment of system performance would have to predict processing time under a variety of conditions. Installation managers need performance information when they configure hardware systems or design data sets. Conversely it may be necessary to plan a GIS application to fit within existing hardware or data constraints.

The performance assessment of commercial GIS has traditionally taken the form of application-specific benchmark tests commissioned by individual user agencies. For the sake of economy, measurements are made on only those GIS functions that the agency intends to use most frequently, and the functions are measured using a limited number of real-world data sets that represent typical workloads for the installation or extremely heavy 'worst-case' conditions. The test designer does not have access to the source code because it is proprietary. Likewise the designer owns the test design, and the results are kept confidential.

Application-specific testing is inadequate to evaluate non-commercial GIS; a diverse set of users needs to refer to the same set of results. All GIS functions in the function set must be evaluated. More important, the testing methodology must enable users to predict performance levels under a wide range of data and operating conditions.

Performance Evaluation Methods

A formal evaluation methodology is used in computer science to analyze and improve computer system performance. It is comprised of a set of quantitative procedures that measure performance in terms of time spent and space utilized in a system. In common practice performance times are measured with internal probes while an application runs on the system. The probes (usually calls to the system clock) may be placed selectively within the program to time specific sections of the code and discover where the system "spends its time".

Performance evaluation studies are applied to entire computer systems, or hardware and software complexes. The interaction between software instructions and the way they are executed in the hardware is system dependent; it is commonly understood that a program cannot be measured outside the context of the computing environment. Program A might run more efficiently than Program B in one environment and less efficiently than B in another environment (Ferrari, 1978).

In a similar fashion the performance of a system may be expected to vary with the application. In this context the term "application" represents a specific workload (in the form of a specific set of tasks and a given data set) that a system is to process. A single evaluation project consists of a number of measured runs which process controlled versions of a synthetic workload. The workload is designed to be modified according to specific parameters; individual performance-influencing factors may be isolated and modified while other characteristics are held constant (DeWitt, 1985; Heidelberger, 1984). This technique can support a full factorial design which measures performance under all combinations of selected factors in an n-tuple structure (Ferrari, 1978).

The complete set of runs tests the strength of the factors as performance predictors. The formulation of test objectives should be based on extensive knowledge of how the system works. Without prior familiarity with the internal organization of the system, an evaluator would not know which facets of the system to test (Heidelberger, 1984).

Purpose of the Study

Compared to the large commercial GIS, the new non-commercial microcomputer-based GIS are uniquely 'testable'. They can be monitored internally because the source code is readily available. An examination of the code also reveals possible performance-influencing factors. There is no reason to limit testing to specific applications or to use only 'typical' data loads, and test results can be made available to the entire user community. In addition, the new PC-based systems use a gridcell structure, and the regularity of gridcell processes would indicate important and highly reliable predictive factors in the data load.

This paper demonstrates the use of performance evaluation techniques (in the form of synthetic data set design, internal monitoring of the code, and the use of a factorial testing scheme) in the setting of the non-commercial PC-based GIS. Tests are conducted on one of the Ohio State University versions of the Map Analysis Package, OSU MAP-for-the-PC. This particular GIS was selected because the source code was already in hand. A method for testing the speed of individual GIS functions is introduced and applied to four of the functions.

Measurable Factors vs. Functional Factors

"Performance" refers to how well a system works. It is based on measurable factors within the system (in terms of the utilization of system resources) and on functional factors such as ease of use, correctness, availability, reliability, training, etc. (usually measured in terms of human resources). In both cases most resources are represented by some form of time expenditure. System resource expenditures might be expressed as throughput or turnaround time; human resource expenditures might be the time required for data collection and editing, or the time required to develop applications. Some functional factors cannot be measured at all; they are simply verifications that the system possesses specified features.

Literature about performance evaluation acknowledges that human resource expenditures are at least as important as computer resource expenditures in evaluating system performance (Stonebraker, 1985). In fact these elements do assume a major role in application-specific GIS benchmark studies, because production schedules are called into play (Goodchild and Rizzo, 1986; Tomlinson, 1981; Greenlee et. al., 1986). However the technical performance literature deliberately excludes the functional factors from formal study because they do not lend themselves to quantitative measurement (Heidelberger, 1984). If performance evaluation methodology is to be used to improve GIS tests, it is more likely to be in the arena of internal measurements of the system itself.

OSU MAP-FOR-THE-PC

Operating Environment

OSU MAP-for-the-PC runs on an IBM PC/XT or PC/AT or PS/2 or equivalent machine, using the MS-DOS or PC-DOS operating system. The machine must have at least 512 KB of memory, a hard disk, and one floppy disk drive. 640 KB of memory is required if grids larger than 28,000 cells are to be used. An appropriate math co-processor (the 8087 for the PC/XT, the 80287 for PC/AT-class machines or the 80387 for "386" machines such as the IBM PS/2 80) is optional but highly recommended, especially with the slower machines.

The microprocessor in the machine -- the 8088 chip in the PC/XT, the 80286 chip in the PC/AT and the 80386 chip in the "386" machine -- determines its processing speed. The pace at which instructions are processed is measured in terms of

a steady beat supplied by a **clock generator**, which beats at 4.77 MHz in the PC/XT, 6 to 20 MHz in the 80286-based machines, and 16 to 25 MHz in the 80386-based machines. Clearly the processing speed of a system has a strong influence on the performance of any GIS application.

Math co-processors may have a strong impact on the execution speed of programs because they vastly accelerate the processing of floating point operations. Co-processors are not essential to program execution, and their installation is optional. In OSU MAP-for-the-PC most mathematical operations are performed in integer arithmetic. However floating point operations are concentrated in the implementation of a few GIS functions, and users who need to call heavily on these functions may find that a co-processor is beneficial.

Ohio State provides two versions of OSU MAP-for-the-PC based on co-processor options. One version requires a math co-processor and the other version emulates a co-processor regardless of the hardware configuration. The emulation version detects the presence or absence of a co-processor and accesses the chip if it is present.

Program Structure

The central module of OSU MAP-for-the-PC is a large command interpreter. After it parses a user command it calls the appropriate subprogram to implement the spatial data handling function that has been requested and then writes the results to the database. The function-handling subprograms can be viewed as independent and unrelated spokes extending from the command interpreter "hub".

The gridcell structure simplifies processing because of its regular distribution of data points and because (for many GIS procedures) all cells must be examined in turn regardless of the cell value. In addition the region boundaries in OSU MAP-for-the-PC must be identical for all the layers. This regularity in the data organization and the data handling processes implies that the number of cells in the grid has a strong influence on performance. It also suggests that the relationship between performance and the number of gridcells is linear.

TEST DESIGN

Functional Level Tests

According to Goodchild and Rizzo (1986) tests should be performed at the level of the spatial data handling function (i.e. they should measure the performance of entire functions) because it is at this level that different GIS packages must be compared. Because they lack a formal command language, the public domain microcomputer-based GIS operate at the level of the "atomic" function (polygon overlay, reclassification, etc.). In the present study each atomic function is subdivided into component segments at the program module level, which is a more detailed level than the one suggested by Goodchild and Rizzo. Modules are measured individually, and the measurements may be summed to find the total measurement for the function.

Performance-Influencing Factors

The performance-influencing factors were chosen to test simple but potentially strong relationships, with a minimum of interaction among the factors. Three factors were selected, one from the workload characteristics and two from the operating environment: the size of the grid, the presence or absence of a math co-processor, and the selection of the microcomputer itself.

The influence of the simple gridcell structure on performance has already been discussed. It is postulated that the volume of data in a given layer (i.e. the size of the grid) would exert an overwhelming influence on the execution speed of GIS functions that process each cell in turn.

The most influential factor in the operating environment appears to be the choice of the microcomputer. The PC/AT, with its 80286 microprocessor and its 8 MHz clock speed is reported to run almost 8 times as fast as the PC/XT with its 8088 microprocessor and its 4.77 MHz clock speed. The presence or absence of a math co-processor was chosen as a factor because the effects of a co-processor were unknown; the program makes little use of floating point arithmetic.

A factorial design was used to test the four GIS functions under a number of factor values. The microprocessor factor was tested in PC/XT and PC/AT 8 MHz configurations. The co-processor factor could be configured as either "on" or "off"; the chip could be either present or absent. Tests on the PC/AT machine were run with and without an 80287-10 math co-processor running at 10 MHz; tests on the PC/XT machine were run with and without an 8087 math co-processor running at 5 MHz.

The workload factor was tested at three levels, represented by grids of 8,000, 16,000 and 24,000 cells. Although the number of levels is too small to support statistical analysis of the results, it is sufficient to suggest a pattern in performance. The small number was used to keep the factorial design to a reasonable size. As it was, the triple of two (microprocessors) times two (co-processor configurations) times three (workload levels) resulted in twelve tests for each of the four functions.

Workload Design

GIS functions were chosen for study based on the way they handled data and based on their ability to illuminate performance-influencing relationships. The GRID function, one of the data entry functions, reads rows of data into memory cell by cell from an external ASCII data file. It also transfers the appropriate data into the four binary data files on the disk. It was chosen because it handles all cells in the same manner, and because it was perceived to have a relatively long duration. This is relevant because the longer running functions cause greater user uncertainty about waiting times.

The SCORE function loads two data layers into memory and derives a complete cross tabulation of their cell values. Like the GRID function, it executes the same process for each cell. In addition it contains a limited number of floating

point operations and it spends considerable time drawing tabular output on the screen. It would be worthwhile to test this output procedure under different conditions. The **MULTIPLY** function loads two layers into memory, performs polygon overlay by multiplying the corresponding cells of the layers, and writes the result to a new layer on the disk. It was selected for the study because it treats all cells the same and because overlay is perhaps the most important class of GIS function. The **CONTOUR** function creates and displays a contour map of a layer in vector mode. It was selected for the study because it relies heavily on floating point calculations and because it can be used to show performance variations caused by the presence or absence of a math co-processor.

Synthetic Data Set

The objectives of the synthetic data set design were 1) to control the size of the grid and 2) to hold constant all other data characteristics as much as possible.

Three databases were constructed with grids of 100 x 80 cells, 200 x 80 cells and 300 x 80 cells, and two layers were created in each database. One contains a series of 80 vertical stripes running the length of the map layer, one cell wide (titled **STRIPE**). The other is a two-color test pattern (titled **ZORRO**) in which the proportion and distribution of the values is the same in each database. A figure "X" extends from the four corners of the layer. It intersects a figure "Z" whose top and bottom bar divide the layer into thirds.

STRIPE presents a situation in which each cell has a different value from the next, and the rows are identical in the three databases. The difference among the three grid sizes lies entirely in the number of rows. This structure insures that each row is processed in an identical manner, and that the maximum number of variations in cell values is presented. **ZORRO** was chosen for its simplicity, for the fact that the pattern extends throughout the layer, and because the distribution of cell values can be reproduced for databases of any size. Among the three databases in the study, the three versions of **ZORRO** are vertically proportional and horizontally identical. In both layers, the effects of several types of data complexity are held constant: the number of different cell values, the distribution of cell values, and the number of horizontal runs in a row.

A third layer, called **ELEV**, was derived from the **ZORRO** layer for the purpose of producing contour maps. The **SPREAD** function of OSU MAP-for-the-PC was used to convert **ZORRO** into a map of distances from the test pattern. The result consists of a set of concentric bands of cells of equal value surrounding the "Z" and the "X" pattern. The value of the cells in each band reflects its distance from the test pattern.

Test Scripts

During each run the program was monitored while it executed a short script of operations. Different scripts were used to test different functions, and in most cases they were comprised of a single command followed by the command to exit

the program. In the script for the GRID function, GRID is asked to initialize the STRIPE layer by reading cell values from a raw data file. The script for the MULTIPLY test requires the multiplication of the STRIPE layer by the ZORRO layer. The SCORE script calls for a cross tabulation report regarding STRIPE and ZORRO. The CONTOUR function is required to produce a contour map of the ELEV layer, specifying ten contour levels in all.

Performance Monitor

The Pfinish* performance monitoring software was used to measure performance times and to count the number of times different sections of the code were entered. Pfinish allows the user to define and measure blocks of code as large as the entire program and as small as a single executable statement. The user may also request a number of output reports which aggregate performance test results. The most prominent report in the profile, which record the number of times each block of code is visited along with the combined duration of the visits. Timing in Pfinish relies on the hardware clock, which has a resolution of 18.2 ticks per second (Phoenix Technologies, 1986).

The user lists the blocks to be measured and requests output reports in a batch file. When a test is run, both Pfinish and the program being tested are loaded into memory. Then the program being tested runs while Pfinish records the performance information that was requested in the batch file, and the appropriate output reports are generated.

The fact that Pfinish is resident introduces certain artifacts. It slows down the apparent processing speed of the program (however this does not affect the internal execution time of the program). Pfinish also occupies space on the disk, and this reduces the available space for the program. During the tests on OSU MAP-for-the-PC, the layer size in the GIS had to be reduced to 25,000 cells in order to accommodate the performance monitor.

PERFORMANCE MEASUREMENTS AND RESULTS

Performance Indices

Blocks were defined at the level of the program module. This decision was based on the fact that each function is associated with its own module (sometimes two or three modules) which does nothing but implement that function. A different batch file was used to test each of the four functions, GRID, SCORE, MULTIPLY, and CONTOUR, and the same batch file was used to manage all twelve tests of a single function. Each batch file was constructed to measure the important modules that implemented the function.

Three modules or sets of modules played a key role in each batch file. One of these was always the main module or modules that implemented the function. The second was the module which transfers blocks of data to and from the disk. The final required module is the one that waits for user input from the keyboard. It had to be included because a method was needed to subtract out the time spent waiting for user input.

Preliminary Results

The entire battery of tests consists of twelve different tests for each of four functions: 48 tests in all. Ordinarily each of these would be run several times and results would be expressed as the means and variances of a number of runs. This procedure is necessary to reduce the noise, or the variation, produced by the clock resolution. Because the current project is limited in both size and scope, multiple runs were not attempted. Instead, the modules that primarily implement each function were run five times together at the outset to give an indication of the amount of variation that could be expected. In all cases there was less than 5% variation.

Results

The analysis of results was limited to scrutiny of a small number of primary modules. It is in these modules that performance-influencing factors can be measured. A list of the modules that implement each function is shown below.

Function	Modules
=====	=====
GRID	INPUTR
SCORE	SCORE, PSCORE, OUTSTR
MULTIPLY	OVRLAY
CONTOUR	THREAD, COTOUR

The graphs in Figures 1, 2, and 3 show the timing results for the GRID, SCORE, MULTIPLY and CONTOUR functions. In each graph a line connects performance times of modules processing the 8,000-, 16,000-, and 24,000-cell workloads. Each line represents a different combination of microcomputer and co-processor factors in the operating environment. All results are measured in clock ticks.

In all the modules, performance in the PC/XT environment appears to be four to five times the duration in comparable PC/AT configurations. The primary interaction among the factors is tied to the machine speed. In all cases the variation associated with the grid size factor and with the co-processor factor is more pronounced in the slower machine environment. The trend is depicted in the form of steeper curves showing the variation due to data volume. It is depicted in the form of greater distance between curves showing the differences due to the presence or absence of a co-processor.

The math co-processor appeared to have almost no effect in either the GRID or the MULTIPLY function, and this is because neither function relies on floating point processes at all. However the CONTOUR function makes extensive use of floating point operations, and the results of the CONTOUR tests show a substantial effect of the co-processor. The THREAD module performed ten times as fast on a PC/XT with a co-processor than on a PC/XT with no processor installed.

Because each line on the graphs represents only three

observations, the lines cannot be analyzed to predict performance for a full range of workloads. However, the graphs indicate that strong linear relationships exist between the size of the grid and the performance time.

SUMMARY

These tests represent a demonstration of testing techniques for microcomputer-based gridcell GIS. The results show promising patterns, but conclusive results would have to be based on a more extensive testing regime. All tests should be run a number of times, and results should be expressed as the means of the output of the runs. In addition, a larger number of observations is needed to properly test the influence of the data size factor.

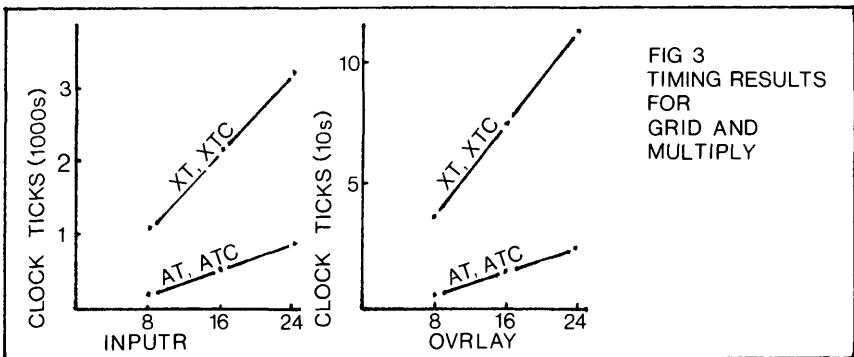
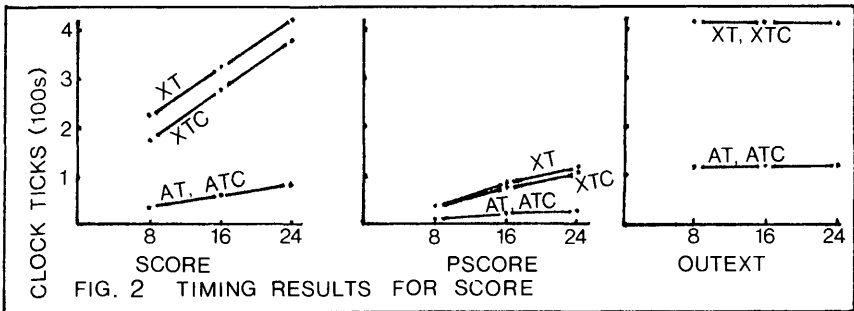
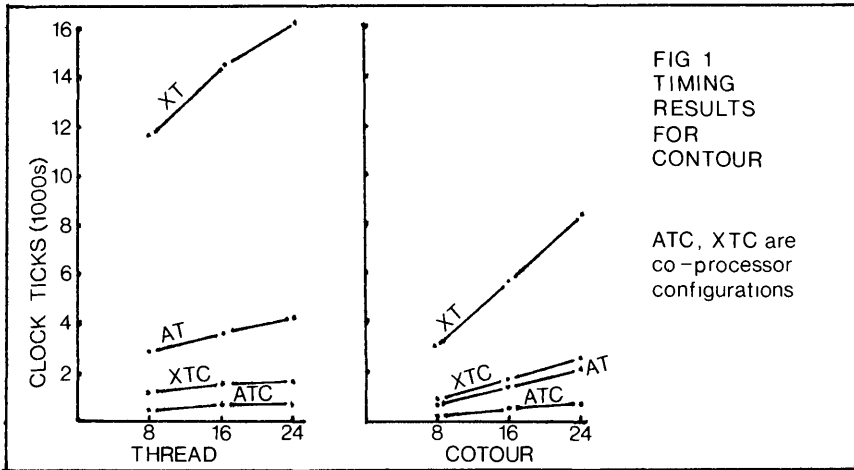
A further variation in the workloads is needed to represent a more complete assortment of predictive factors: the variety of distinct cell values, the number of horizontal runs, the probable distribution of cell values, user-defined search radii, etc. The results of such a study can be analyzed using simple regression techniques. Outcomes for individual modules might then be summed to arrive at performance predictions for entire functions.

BIBLIOGRAPHY

- Brickner, R. G., 1986. "An Execution Profiler for the PC," PC Tech Journal, Vol 4 (11): 120-130.
- _____, 1987. "Execution Profilers for the PC, Part 2," PC Tech Journal, Vol 5 (2): 166-171.
- DeWitt, D. J., 1985. "Benchmarking Data base Systems: Past Efforts and Future Directions," Data base Engineering, Vol 8 (1): 2-9.
- Ferrari, D., G. Serazzi and A. Zeigner, 1983. Measurement and Tuning of Computer Systems. Prentice-Hall, Englewood Cliffs, N.J.
- Goodchild, M. F. and B. R. Rizzo, 1986. "Performance Evaluation and Workload Estimation for Geographic Information Systems," Proceedings, Second International Symposium on Spatial Data Handling.
- Greenlee, D. D., J. W. Van Roessel and M. E. Wehde, 1986. An Evaluation of Vector Based Geographic Information Systems at the EROS Data Center, U.S.G.S. EROS Data Center, draft report.
- Heidelberger, P. and S. S. Lavenberg, "1984. Computer Performance Evaluation Methodology," IEEE Transactions on Computers, Vol c-33 (12).
- Marble, D.F. and L. Sen, 1986. "The Development of Standardized Benchmarks for Spatial Data base Systems," Proceedings, Second International Symposium on Spatial Data Handling.

Stonebraker, M., 1985. "Tips on Benchmarking Data Base Systems," Database Engineering, Vol 8 (1): 10-18.

Tomlinson, R. F. and A. R. Boyle, 1981. "The State of Development of Systems for Handling Natural Resources Inventory Data," Cartographica, Vol 18 (4): 65-95.



DATABASE SIZE (1000s of cells)

USE ERROR: THE NEGLECTED ERROR COMPONENT

Kate Beard
Department of Surveying Engineering
University of Maine-Orono
Orono, Maine 04469

ABSTRACT

Commonly recognized map errors include those associated with data collection (source error) and the processing of data for map compilation (process error). Another error component, use error is defined and added to this typology. This paper argues that without attention to use error, large investments to reduce source and process error may be wasted. Traditional representation of spatial information on paper maps has limited our ability to control this form of error in any significant way. While the misuse of maps cannot be entirely avoided, computer technology offers a possibility for limiting the opportunities for misuse. This idea is explored by examining ways in which maps are misused, and from this exploration, formulating geographic information system design strategies that may counteract the potential for use error.

INTRODUCTION

"We must not forget that like carpenter's tools, maps should not be misused. More should not be expected of them than they can perform." (Wright 1942 p. 593).

Despite warnings such as Wright's, maps are often misused. We readily recognize misuse, but treat it as unavoidable. Map producers are not held directly accountable since they can have no certain knowledge of their audience or how their products will be utilized (Jenks and Caspall 1971). The conscientious producer attempts to control misuse by maintaining scrupulous quality control during production, and hopes that once the map goes out for distribution, it *will be used in a reasonable manner*. This can be a losing battle of larger investments in quality control with little or no assurances of reduction in misuse. The advent of geographic information systems has promised improvements in spatial data handling and analysis, but GIS has the potential to fall into the same trap; better quality control but no insurance against misuse. We cannot assume that GIS

will automatically be less susceptible to misuse than traditional maps, and it may, in fact, exacerbate the problem by expanding access to mapped information.

Misuse of maps can have serious repercussions particularly when the end result is some legislative action. Efforts in quality control help indirectly, but misuse requires more direct attention. The development of GIS provides an opportunity to directly address misuse. Because users must interact with a system to use spatial data, a GIS can be consciously designed to avoid or minimize misuse. This paper provides a preliminary exploration of this idea. It examines the need for greater attention to the misuse of maps, explores the nature of map misuse, and considers strategies to avoid misuse through the design of GIS.

A TYPOLOGY OF MAP ERROR

This section considers the misuse of maps in the larger context of map error. Errors in maps can be contributed by any number of factors. Data collection is the first phase in which errors can be introduced and the term source error will be used to describe these errors. Source errors can include errors in the positional description of the data or in the identification and discrimination (Chrisman 1982) of spatial objects. Limitations in data collection instruments, negligence on the part of the collector or instrument operator, adverse weather conditions, time constraints and other variables can contribute to the source error component. The cost of data collection and available funding, while not directly contributing to error, influences the precision, accuracy and completeness with which spatial information can be collected. The term source error, in this case, includes errors in completeness and positional and attribute descriptions introduced during data collection.

Manipulations of the data subsequent to collection, such as digital conversion, generalization, scale change, projections, and graphic representation can introduce additional errors. These errors will be referred to as process errors. In traditional map production, the occurrence of process error generally ceases with the final compilation and publication of a map. In a digital environment, the potential for process error is always present since manipulations are easily carried out and each step potentially contributes new errors to the data.

In general, surveyors and cartographers share a concern for minimizing source and process errors. The analysis of errors in

spatial data collection is a fundamental part of the surveying sciences, and cartographic training emphasizes faithful depiction of data in the transformation to map form. Conformance to professional standards, careful calibration of data collection instruments, and more accurate instruments may help to reduce source errors. In fact, we generally assume that these errors can be corrected by larger budget outlays, better instruments and more rigorous specifications of quality control. Likewise, increased precision and resolution in hardware devices, and quality control in software production are expected to reduce process errors.

Reductions in source and process errors improve the overall quality of a map and its usefulness. The correctness of a map, however, provides no guarantee that it will be correctly used. As Gersmehl (1981,1985) points out, the potential for error does not end with the compilation and publication of a map, but is attached to the very existence of a map and its use. The map itself is static (the published map accumulates no new source and process errors), but its existence and duration over time create the possibility for use errors and an increased probability for errors with the passage of time. Use error in this case will refer to the misinterpretation of maps or the misapplication of maps in tasks for which they are not appropriate.

Use error is typically not recognized as a component of map error. Unlike the case of source and process errors, no professional group or discipline directly addresses use error. Also no formal training is assumed necessary for map use. As Keates (1982) suggests, many users would maintain that using a map requires no more than normal vision and average intelligence. Errors in map use, however, can carry significant penalties, since a single case of misuse can cancel all investments in source and process error reduction. Failure to consider use error in the past was excusable, but failure to consider it now risks many of the benefits we hope to achieve through GIS.

Misuse of maps has received little systematic study. We can point to specific cases of misuse, but we currently lack a comprehensive understanding of how and why maps are misused. Discovery of common characteristics in misuse can lead to a strategy for corrective action through GIS design. The next section examines cases of map misuse.

USE ERROR

Misuse of maps can occur in several ways. It is not possible to

exhaustively document all instances of misuse, but a few examples help to illustrate the range of cases. Gersmehl (1985) cites an instance in which he compiled a dot map of histosols (organic soils) of the United States. Each dot was used to represent the general location and size of a histosol occurrence, except three, which Gersmehl confesses to placing somewhat spuriously. This map appeared some years later reinterpreted as a map of Peatlands of the U.S. All of the dot locations, including the three spurious dots, were designated on the map as major peat deposits.

Two misuses of Gersmehl's histosol map are demonstrated by the Peatlands map. The dots, in two cases, were intended as generalized symbols of a few small, and widely scattered occurrences of histosols. On the Peatlands map these were depicted as sizeable peat deposits corresponding with the locations of the original dots. The other misinterpretation was that all histosols were assumed to be peat deposits, (not a correct assumption). This fact was, of course, well known to Gersmehl, but was nowhere communicated on the map.

We can identify at least two generic causes for these cases of misuse: lack of information and divergence from convention or expectation. Physical space limitations and graphic conventions restrict the amount of information which can be shown on a map. Gersmehl's choice of scale limited his ability to present a more complete description of the information. If additional attributes of each histosol type had been included, such as its peat potential, the error might have been avoided. His choice of scale also forced him to sacrifice positional accuracy for graphic emphasis, an instance of cartographic license which lead to nasty repercussions. If users have certain expectations about mapped information, then violations of these can result in errors. Many users assume that the location of an object on a map bears some relationship to the object's true position on the ground. Gersmehl, because of cartographic license in placement of a dot on a map, violated the assumption and introduced the possibility for error.

The Gersmehl case also illustrates that misuse frequently occurs when maps compiled for one purpose are used for other purposes for which they are not suitable. This can happen for a number of reasons; some intentional and some not. Time and budget constraints are common culprits in these cases. Such constraints can prevent the acquisition of appropriate data for the intended use and force the use of available but inappropriate data. In another example, Napton and Luther (1981) note the misuse of generalized

soil productivity maps. The productivity maps were created by aggregating soils data into categories based on yields of corn per acre. To simplify the spatial complexity of the maps, several small adjacent but dissimilar soil map units were combined to create larger productivity units which could be represented at a smaller scale. The final maps did not in any way document the presence, size, shape, quality or location of soils within productivity units which had quite dissimilar productivity levels. Based on comparison with detailed information, the generalized maps were determined to have twenty three percent error in misclassification.

Although these maps were only compiled for very general planning purposes, their concise form promoted their subsequent use for prime farmland designation, zoning administration, and tax assessment. In these cases, the generalized productivity maps became the basis for legislative action with implications for individual property rights and taxes. As Napton and Luther state:

"Maps with this amount of error might be helpful for some purposes, but the existence of the map invites use for many other purposes, ... the employment of this information for local or site specific planning opens the door for court challenge." (1981 p. 178)

This problem can be compounded in the case of digital files. Since digital files are still time consuming to create, uses of existing files can be overextended. Blakemore (1985) describes a file of British districts digitized by the Department of the Environment as a thematic base for choropleth mapping. The file had no information on the accuracy of the coastline or internal positional accuracy, yet it was used and misused for many different purposes simply because it was readily available in digital form. Many early digital files were generated from small scale maps since these could be converted most quickly and required the least storage. These maps have limited use for detailed analysis, yet the temptation to use these files remains since they are available.

Legislative mandates can be particularly guilty in this respect by setting timelines which make collection of the appropriate data impossible. In order to meet legislative requirements, any available data is used whether it is suitable or not. The State of Maine recently passed comprehensive planning legislation which illustrates this problem. The legislation requires communities to develop comprehensive plans and subsequent zoning ordinances or other enforcement mechanisms by as early as 1991. Information at

a level of detail sufficient to develop adequate and defensible plans is not currently available, nor likely to be by the legislative deadline. The potential for misuse of existing information is therefore substantial.

There are many opportunities for misuse of available data. Misuse can occur if the available data is out of date, if the scale or resolution of the data is too coarse for the intended application, or if the classification and interpretation of the available data does not support the intended application. In many of these cases we can point to misuse of generalized maps as a common error. Generalized maps are particularly troublesome because they are more restrictive of information and users are often unaware of how much information has been lost. Generalized maps are usually not accompanied by information on the source material, classification and interpretations made during generalization, and the degree of generalization. Without this information, users can quite easily use the data for purposes not originally intended.

Other instances of use errors occur when maps are used for quantitative analysis without recognizing the effects of map scale, conventions, or data type. Measures of point location, of length, area and count vary with changes in scale. Several mathematicians and cartographers have discussed the difficulties of making reliable quantitative measures of phenomena from maps (Steinhaus 1954, Richardson 1961, Maling 1968, Perkal 1966). Boesch and Kishimoto (1966) also cite the difficulties of making reliable counts of objects from maps. Using maps to make counts of objects leads to errors if the completeness and currency of the maps have not been accounted for. Making counts from complex maps is also a case in which visual processing is not efficient. Often the level of measurement (Stevens 1946) of the data represented on maps is not accounted for in analytical use of maps. Hopkins (1977) points to the addition of ordinal valued maps as a common error in suitability analysis. Others have presented the errors associated with the overlay of maps for planning purposes (MacDougall 1975, Chrisman 1982, 1987).

The above examples describe cases in which maps were an appropriate representation medium for information, but were inappropriately used. A different case of misuse arises when a map itself is the wrong medium for presenting information. An example of this misuse is illustrated in *Zinn v. State of Wisconsin*. A hearing examiner highlighted a contour on a USGS quadrangle map to indicate legal evidence of the ordinary high water mark (OHWM). Land below

the ordinary high water mark belongs to the state. Based on the map evidence an owner of land abutting the lake asserted that the state had claimed most of her property, created a cloud on the title to her land, and deprived her of her riparian rights to the lake. Epstein and Roitman (1987) suggest that the graphic depiction of the OHWM on the map provoked the legal conflict. A direct statement that the OHWM existed at elevation 990 would have been preferable. Presentation of the information in this manner would have avoided the misinterpretation resulting from the graphic depiction and possibly avoided the conflict.

From these examples we can identify several common causes for map misuse which can be summarized as follows:

- Lack of information.
- Deviation from conventions and expectations
- The use of small scale, generalized maps for many uses because they are convenient and less expensive.
- The lack of current data and ability to make frequent updates.
- A lack of documentation on data quality

The presentation of spatial information in map form demands data reduction. Early computer systems also suffered from limited storage capability. Decreases in the cost of digital storage and increases in the speed of digital processing remove some of these barriers. Advances in digital mapping will now allow greater control over use error than existed with paper map production. The next section suggests that some of the generic cases of map misuse can be avoided or mitigated by specifically designing systems and databases to avoid them.

RECOMMENDATIONS FOR CONTROLLING USE ERROR

The above examples indicate that omission of information due to physical constraints, generalization, lack of currency, and lack of quality documentation are primary contributors to use errors. Recognition of these as common contributors to misuse can lead to solutions. Without attempting to predict the potential misuse of different spatial data sets, we can nevertheless guard against the possibility by improving spatial information management through GIS design. Some directions for system design which show promise

include:

The ability to store more information than was previously allowed by physical map sheet size.

Often much more information is collected during inventory than is passed on to eventual users. This restriction on dissemination of information has been due to the physical limits of the paper map. Early computer systems were restricted by limited storage space, but such limitations are rapidly disappearing. A digital data base can now be structured to store more information for access by users.

Representation of more detailed, disaggregate data

Generalized maps have been subjected to summaries, aggregations and other reductions of information for specific purposes. Detailed, disaggregate data would give users the flexibility to aggregate the data for their specific needs. Users would not be constrained by previous summaries or interpretations (aside from potential biases in data collection) which could impact their questions. Any generalization or aggregation of the detailed data requested by users should be documented so that the extent and location of data modifications would be available to subsequent users.

Potential for more extensive data quality documentation

Paper maps may include reliability diagrams or other sketchy information on data quality. Digital databases provide the potential to associate quality information with individual objects and their descriptions. As an example, Dutton's (1983) GEM structure can represent positional accuracy by the depth with which an element is placed within the structure.

Improvements in updates to maintain currency

Paper maps are often out of date because publication costs limit frequent reissues. Use of out of date maps lead to errors. Digital databases have the potential to support more frequent updates although real time transactions on spatial data have not been perfected. Data documentation should specifically include information on currency.

Structuring data to avoid illegal or illogical operations

Certain mathematical operation are only valid for certain levels of information. For example addition and subtraction of nominal or ordinal valued data is meaningless. Databases can be structured so only valid operations can be applied to particular data types. A database might also be designed to detect when the resolution of the data is insufficient for a particular application.

CONCLUSION

Although misuse of maps in the context of paper maps was an insoluble problem, developments in GIS have the potential to overcome many cases of misuse. If we are to reap the full benefits of GIS, we should not overlook this opportunity to include use error in overall plans for improved quality control.

REFERENCES

- Blakemore, M. 1985. 'High or Low Resolution? Conflicts of Accuracy, Cost, Quality and Application in Computer Mapping,' **Computers and Geoscience**. 11: 345-348.
- Boesch, H. and Kishimoto, H. 1966. **Accuracy of Cartometric Data** Zurich: Geographisches Institut.
- Chrisman, N. R. 1982. Methods of Spatial Analysis Based on Error in Categorical Maps. Unpublished PhD thesis, University of Bristol.
- Chrisman, N. R. 1987. 'The Accuracy of Map Overlays: A Reassessment,' **Landscape Planning**. 14: 427-439.
- Dutton, G. 1983. 'Geodesic Modeling of Planetary Relief,' **Proceedings AUTO CARTO**. 6 2: 186-201.
- Epstein, E. and Roitman, H. 1987. 'Liability for Information,' **Proceedings URISA**. p. 115-125.
- Gersmehl, P. J. 1981. 'Maps in Landscape Interpretation,' **Cartographica**. 18: 79-109.
- Gersmehl, P. J. 1985. 'The Data, the Reader, and the Innocent Bystander,' **The Professional Geographer**. 37: 329-334.
- Hopkins, L. 1977. 'Methods of Generating Land Suitability Maps,' **American Institute of Planning Journal** 386-400.
- Jenks G. and F. Caspall 1971. 'Error on Choroplethic Maps: Definition, Measurement, Reduction,' **Annals, Association of American Geographers**. 61: 217-244.

- Keates, J. S. 1982. **Understanding Maps**. New York: John Wiley & Sons.
- Maling, D. 1968. 'How Long is a Piece of String,' **Cartographic Journal** 5: 147-156.
- MacDougall, E. B. 1975. The Accuracy of Map Overlays. **Landscape Planning**. 2: 23-30.
- Napton, D. and Luther, J., 1981. 'Transferring Resource Interpretations: Limitations and Safeguards,' **PECORA VII Symposium**. 175-186.
- Perkal J. 1966. 'On the Length of Empirical Curves,' Trans. R. Jackowski. Michigan Inter University Community of Mathematical Geographers. **Discussion Paper No. 10**.
- Richardson, L. F. 1961. The Problem of Contiguity,' **General Systems Yearbook**. 6: 139-187.
- Steinhaus, H. 1954. 'Length, Shape and Area,' **Colloquium Mathematicum**, 3: 1-13.
- Stevens, S.S. 1946. 'On the Theory of the Scales of Measurement,' **Science**. 103: 677-680.
- Wright, J.K 1942. 'Map Makers are Human,' **Geographical Review**. 32: 527-544.

EXTENDING ENTITY/RELATIONSHIP FORMALISM FOR SPATIAL INFORMATION SYSTEMS

Dr. Yvan Bédard, Director
François Paquette, B.Sc.A., M.Sc. candidate
Laboratory for Spatial Information Systems
Geomatics Center
Dept. of Geodetic Sciences and Remote Sensing
Laval University
Sainte-Foy, Qc
Canada, G1K 7P4
(418) 656-3694
BITNET 1130025@LAVALVX1

ABSTRACT

Information Engineering develops and uses systematic methodologies and tools to facilitate the implementation of information systems. However, for Spatial Information Systems (SIS), these methodologies and tools need to be extended to better consider the spatial characteristics of the data and of their processing. One of these tools, the Entity/Relationship (E/R) formalism, is more and more used to build data models for SIS. This paper describes difficulties of the standard E/R formalism with regards to spatial phenomena and suggests three extensions to improve E/R effectiveness: the Sub-Model Substitution (SMS) technique, the inclusion of *cartographic only* objects, and generalization. This paper also provides additional research directions to improve E/R formalism expressive power.

INTRODUCTION

Over the last three years, we have seen more and more papers on the use of E/R modeling for Spatial Information Systems (SIS). Although this is a useful tool for the implementation of SIS, its expressive power is somewhat limited for spatial phenomena. Specialists in non-spatial information systems (e.g. banks, hospitals, schools) have been working with E/R for the last ten years and have recognized the need to improve E/R modeling for special purposes.

This paper begins with an overview of what data models are and what is their "raison d'être". Then come definitions and rules to build conceptual data models: the Individual Formalism, also called French E/R. Some problems related to the modeling of spatial data with such a formalism are presented. Then, it is used with three extensions to better model spatial phenomena: the Sub-Model Substitution (SMS) technique, the inclusion of *cartographic only* objects, and generalization. We close this paper with practical results and additional research issues.

SPATIAL DATA MODELING

Data models are simplified views of a part of the reality, they are built according to certain rules to facilitate the implementation of a database in an information system. Shlaer and Mellor (1988) mention that a data model is "a *thinking tool* used to aid in the formalization of knowledge". In fact, our general capability of understanding, remembering, making decisions and communicating depends upon our capability of making models.

Data modeling is an abstraction process where the essential elements are emphasized and the non-essential ones eliminated with regard to a specific goal (e.g. improve transportation, provide better management of property files). Data modeling requires the use of rules to create the model (e.g. Codd's normal forms) and to communicate this model, i.e. a language using a well-defined set of symbols (literal and/or graphical) with associated meanings.

Building good data models is very important since they play a major role in the determination of "which part" of the reality is being represented in the database, how it is represented, what can be done with this representation, and how fast it can be done. In addition, data models describe the most stable and expensive resource of an information system: data.

Creating data models is a multi-step mapping of the reality into a physical database and its representation to the users. Well-known examples are the three types of data models (internal, conceptual, and external schemas) identified by the ANSI/SPARC Study Group on Data Base Management Systems (1975). The **Conceptual Schema** is a representation of the reality showing all entity types to be included in the database, their attributes and their relationships. This view is independent of the type of Data Base Management Systems (DBMS) used. It is written in a simple language and is directed towards the system manager. From the programmers' point of view, this is a high level of abstraction of the database structure. At this level of abstraction, the expressive power of semantics formalisms such as E/R is necessary.

The **Internal Schema**, also called physical schema, shows the view of the reality as it is structured in the computer database, i.e. how data are physically stored and related. It usually is written with the Data Definition Language (DDL) of a DBMS or with standard programming languages such as PASCAL, FORTRAN or COBOL. Thus, the internal schema depends upon the software or language used for the implementation of the information system. This data model is written in a more complex language (programming code) adapted to programmers and computers. From the programmers' point of view, this is the lowest level of abstraction of the database structure. This data model can be a direct translation of the conceptual model or one optimized for better computing performance.

The **External Schema**, also called user schema, is an exact or modified subset of the conceptual schema. It is built to illustrate which entity types, attributes and relationships of the database are available for a specific use or user. It usually is written in the same simple language than the conceptual schema and is directed towards end users for specific applications. From the programmers' point of view, this is the highest level of abstraction of the database structure.

More and more, we use an additional data model to ease the translation from the conceptual schema to the internal schema: the **logical data model** (Bédard 1988). This logical schema depends on the type of DBMS used and is written in DBMS-oriented formalisms such as CODASYL and Relational.

Thus, to avoid the difficult task of going directly from "talking about" the reality we want to manage with the information system to "programming" its corresponding database, *different data models are needed at different levels of abstraction*. Such intermediary steps are especially useful when building large databases (as in most SIS). Sometimes, the term "datalogical" is used for the internal model and the term "infological" for the conceptual and external models; this indicates that the latter more closely represent the reality while the former rather represents the database.

Good data models

A good data model includes all the entity types that we want information about, all the attributes necessary to describe the desired characteristics of the selected entity types, and all the necessary relationships among these entity types.

In addition, a good data model eliminates redundancy. This can be done with Codd's (1972) Normal Forms which are well known by relational DBMS modelers. Redundancy elimination is usually done at the conceptual level, leaving the optimization of the data structure (to improve computing performance by reintroducing well chosen redundancies) for the internal model.

Data dictionary

To completely describe the reality (in a database sense), the data models must be completed by a data dictionary. Such dictionary contains all the necessary metadata about the data models. Usually, it includes the name and definition (including examples and exceptions if necessary) of the entity types and attributes included in the data model. It also includes the type (e.g. real, boolean) or each attribute as well as some integrity constraints (e.g. domain of values) and the measurement units used. The programmer should find all he needs to do his task in the data model and the data dictionary.

ENTITY/RELATIONSHIP FORMALISM AND ITS APPLICATION TO SPATIAL PHENOMENA

The objective of E/R modeling is "to create a description of the semantics of data that reflects the actual enterprise and its information requirements" (Martin and McClure 1988). These authors also add that "the task of the data modeler is to capture reality and communicate about it accurately. He tends to be distracted from this task if he has to think about computer hardware or data-base software or if the line between semantics and the implementation of data becomes blurred".

According to the E/R concept, we make conceptual data models by identifying, classifying, describing and relating parts of the real world to organize the information into a formal structure amenable to a computer form. Thus, it is useful to perceive the reality as containing "entities" or objects, "attributes" or characteristics of the objects, and "relationships" between entities:

Entity: object, person, concept or event about which we want information; the type of an entity is usually identified by a noun (e.g. entity type *River*, entity type *Road*).

Relationship: association between two types of entities; usually identified by a verb or a preposition (e.g. *Road to cross River*). A relationship has a **cardinality** giving the number of times (minimum and maximum) the relation can occur between two specific entities (occurrences). For example, if we say that a Road crosses a River a minimum of 0 times and a practical maximum of 5; and on the other hand that a River can be crossed by a minimum of 0 Roads and an unknown maximum of N, this leads to a relation *to cross* with a cardinality of 0,5 in one direction and 0,N in the other direction.

Attribute: characteristic of an entity type or a relationship; for each entity, it contains a value called data; mostly identified by an adjective, a noun, or a group of nouns and/or adjectives (e.g. for Road: *pavement, number of lanes, number of accidents*). When an attribute is used to identify a specific entity (occurrence) within its group of similar entities, this attribute is called **identifier** or key (e.g. for Road: name).

These three basic constructs of the Entity-Relationship model have been graphically represented several ways in the past few years, leading to different E/R representations of the same data model. Several examples can be found in the literature, for example Martin and McClure (1988) have identified three notation styles, each of them different from the original E/R style presented by Chen (1976) or from the styles used in the other references of this paper. For this paper, we use the French notation called "Individual Formalism".

According to this formalism, an entity type is represented by a rectangle containing its name at the top in uppercase letters. A relationship is represented by a line with a central ellipse containing its name at the top in uppercase letters. Attributes use

lowercase letters and are included either in the rectangle of the entity type they describe or in the ellipse of the relationship they describe. Attributes serving as entity identifiers are placed first in the list of attributes and are underlined. All entity names must be unique. The cardinality in one direction of the relationship is placed on the relation line, close to the entity from which we start reading the relationship. (for more information, see Collongues and al 1986, Tabourier 1986, or Tardieu and al 1986)

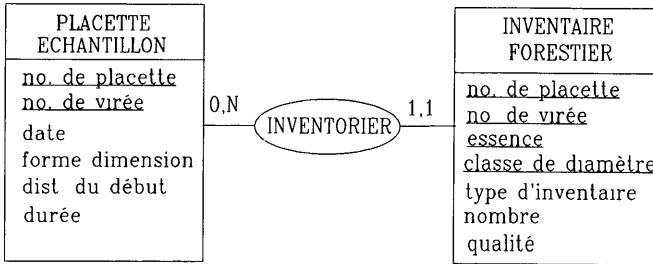


Figure 1: Individual Formalism diagram

This formalism, like other semantics formalisms, is built for a traditional database structure and is not well suited for spatial referencing (location and geometric description of entities) and SIS. For example, it does not easily allow the inclusion of primitive and complex geometric entities (e.g. points, edges, polygons, point sets, nodes, lines, networks, sets of polygons) with spatial attributes (e.g. coordinates, minimum bounding rectangle) and spatial relationships (e.g. connectivity), and the analysis of spatial operations (e.g. overlay, buffering, distance and area measurements, connectivity, intersection, spatial querrying). In addition, not all computerized cartographic objects can actually be represented.

Some problems and considerations

The actual way to deal with spatial referencing is either to avoid its modelization or to add it to the conceptual data model exactly like other entities, attributes and relationships. Avoiding the modelization of spatial referencing creates a conceptual data model not showing all the data available in the GIS database(s). This is a problem per se since the data model must show all entities, attributes and relationships to be included in the global database of an organisation (either located in one or two different databases, graphic and non-graphic, in the GIS). We think that the data model should show which geometric entities are needed (e.g. points, lines, polygones, sets of points) to draw non-geometric entities (e.g. roads, rivers, houses). Knowing the needed geometric entities influences the choice of the GIS system to buy as well as the physical structure of the database. Also, the programmer needs to know all cartographic attributes to include in his program code (e.g. symbology, different geometric descriptions for different scales). All this information can be included either in the data model or the data dictionary.

On the other hand, including information about spatial referencing in a conceptual schema with the actual E/R formalism introduces other problems. The first one is related to the size of the conceptual schema. Very often, SIS are complex systems already involving a large number of entities and relationships and many more attributes, leading to data models complicated to draw, verify, modify, and read. The simple addition of geometric entities (with their attributes) and their relationships (1) among them, and (2) with the non-geometric entities, rapidly fills up the conceptual model, worsening the size problem. For example, every spatially referenced entity has a relationship with the geometric entity describing its shape. With such a solution, the information is available but slow to find because we must

navigate in an already large conceptual model.

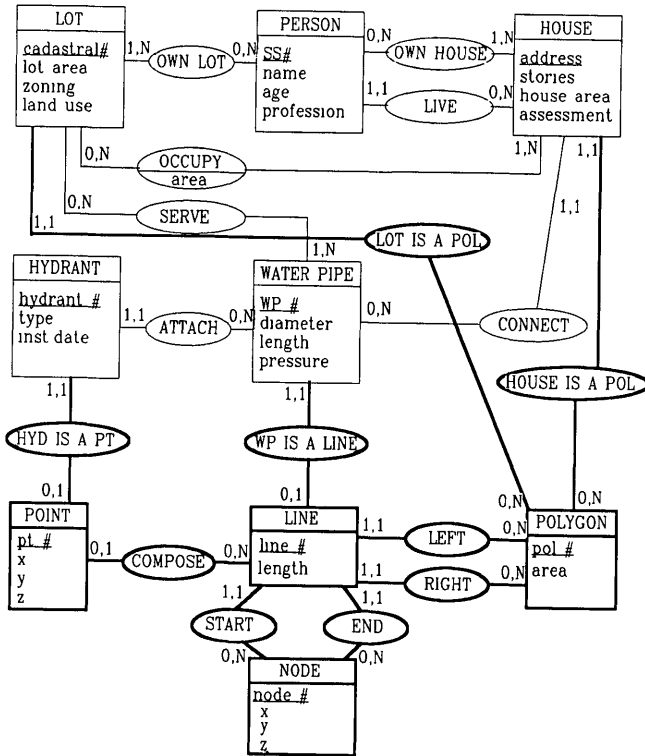


Figure 2: explicitly including (in bold) geometric entities and relationships (from Bédard 1988)

In addition, including the geometric entities in the conceptual model raises the question of "which geometric data structure to use?". This simple question brings up a whole new problem that GIS researchers have worked on for years (e.g. spaghetti, CAD/CAM type structures, topology, Voronoï Diagrams, Cell graphs). We think that this is a problem for GIS developers and scientists and that it should be kept away from GIS users while modeling their database. Furthermore, since one difference between DBMS and GIS systems is that DBMS have "no predefined data structures" while GIS provide "built-in data structures" for geometry, why go into the details of modeling a geometric data structure?

It also is interesting to note that for most GIS users, geometric entities are not real life objects and have no meaningful attributes. Since they are perceived differently than other entities, why treating points, lines and polygons as we treat buildings, roads, and forest cells in the data model? On the other hand, their geometric description is as important as their non-graphic attributes. So, why not indicate the geometric shape of an object with the other attributes, i.e. in the data model?

There is an additional problem with traditional E/R modeling: only the entity types explicitly defined in the database are represented, i.e. only the objects having their own object file containing attributes. This represents a limitation for GIS objects appearing on a map but not needing an object file in the database. These objects appear on maps either as background information or as objects to manage based only on their geometric properties, cartographic layer or symbolic value (e.g. road

types on topographic maps). At first sight, these objects may not be perceived as entity types, but if we consider the conceptual data model as a thinking tool (1) to define what information is available and (2) as a necessary step helping the programmers' task, and if we remember what an entity is, then these objects can be treated as entities.

Furthermore, we are investigating if the conceptual data model should describe *all* the information of an organisation and not only the information to be computerized. Actually, it seems reasonable to think so, especially for SIS which are large systems where not all cartographic information is necessarily computerized. We would then need further extension to E/R.

EXTENDING E/R FORMALISM

Data modeling is still evolving and specific improvements are needed in specific fields, leading to specific data modeling techniques. In a recent research project, we found that the major weakness of actual information system design methodologies, when applied to SIS, is their modelization tools (Boutin 1988). For SIS applications, we need an extended formalism allowing us to include semantics specific to our purposes and to better represent all entities, how they are located, geometrically defined, and spatially interrelated. Such semantics must be included either in the data model or its corresponding data dictionary. Brodie (1984) mentioned that a new generation of data models is emerging: special purpose data models (for applications such as VLSI, CAD/CAM and Cartography). The ideas presented hereafter (SMS technique, including *cartographic only* objects, and generalisation) are still in development but represent a step in the building of such a new generation of formalisms for spatial data modeling.

The Sub-Model Substitution (SMS) technique

As previously mentioned, one of the actual problems with basic semantic modeling such as E/R is that their models rapidly become complex for large databases, making it easier to leave redundancies, inconsistencies and omissions out. Most of this can be solved by a good CASE program (Computer-Assisted Systems Engineering) providing data modeling tools; however, the difficulty involved in reading large models remains and this only is with more powerful modeling rules that readability will improve.

The SMS technique uses a set of meaningful graphical symbols to replace the relationships between the non-geometric entity types of the database (e.g. roads, rivers, forest cells) and the geometric entity types describing their cartographic shape (e.g. lines, networks, polygons). The SMS technique is built to encourage simplicity of building, verifying, modifying and reading.

SMS Rule: if an entity type has a geometric description, the original geometric description is represented by a graphical symbol placed next to the name of the entity type, on its left hand side. Each graphical symbols of figure 3 is a substitution of its corresponding sub-model.

The list of geometric descriptions presented in figure 3 still is in development, however we can already see advantages to the SMS technique: (1) to almost eliminate one relationship per entity type, and (2) to eliminate the need to create a geometric data structure, thus (3) to eliminate all geometric entity types with their attributes and interrelationships. In a data model of medium complexity (46 entity types, 95 relationships, 167 attributes) for a forest application, this has resulted in a reduction of 47 relationships plus a few entity types and attributes (Paquette 1988). The simplicity and usefulness of the SMS technique has already been accepted by two major SIS consulting companies in Québec who have begun to use the idea.









SMS Symbol	Geometry	Sub-model substituted by symbol	
	point	1,1	0,1
	line	1,1	0,1
	polyline	1,1	0,1
	simple network	1,1	0,1
	simple polygon	1,1	0,1
	complex network	1,1	0,1
	joint polygons	1,1	0,1
	partition	1,1	0,1

Figure 3: SMS symbols with their substituted sub-models.

Including cartographic only objects

As previously mentioned, traditional E/R modeling does not accept objects appearing on a map but not needing an object file in the database. This can be solved by the simple rule presented in the next paragraph and which completes very well the SMS rule both in content and readability.

Cartographic Only Objects Rule: if an entity type has non-geometric attributes (e.g. Road: pavement, number of lanes, number of accidents) explicitly stored in the database, i.e. not deduced from the symbology of the base map, then a *database* or *hard disk* symbol must be placed next to the name of the entity type, on its right hand side.

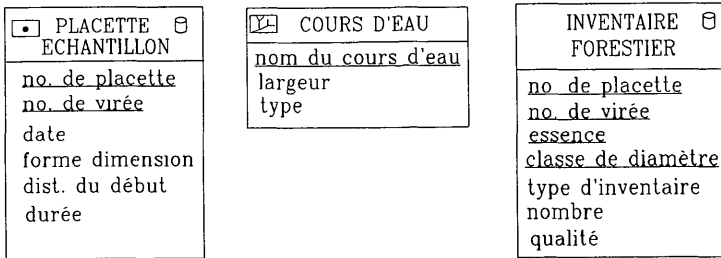


Figure 4: inclusion of the database symbol: (1) a mapped entity type with DBMS attribute file, (2) a mapped-only entity type, and (3) a non-mapped entity type.

This rule allows us (1) to add in our conceptual data model those useful objects which are on a map but which need no database files per se, (2) to clearly differentiate between entity types which are *mapped only* and those which are *mapped and stored in the database as entities*, (3) to know this before building the next data models (logical and internal) which need this information, and (4) to help in the definition of background information for map presentation.

Generalisation

Generalisation allows us to create meaningful groups of entities with common characteristics (e.g. forest operations: spraying, cutting, soil preparation, thinning out, planting, etc.). This is not a new idea in semantics formalism, and it has already

been introduced in the field of SIS (cf. Frank 1985, Blais 1987, Egenhofer 1987). However the following paragraphs state the necessary rules to integrate generalisation with the previous two techniques:

Generalisation Rule 1: a super-entity type can be created from a logical group of entity types all sharing one or more common attributes with common domains of possible values; each common attribute becomes an attribute of the super-entity type. This rule indirectly states that (1) two attributes with different domains of possible values are considered different and must have two different names, and (2) all sub-entity types have, in addition to their own attributes, the super-attributes;

Rule 1.1: for the identifiers, there is the additional condition of unicity among generalized entities; often this may lead to extend the domain of values (e.g. sequential numbers).

Generalisation Rule 2: a super-entity type can only have relationships common to all sub-entity types; when such a relationship exists, all sub-entity types are logically (but not graphically) connected to this relationship. When such relationship does not exist, the relationships directly go to the sub-entity types. This rule indirectly states that a sub-entity type has all the relationships of its super-entity type.

Generalisation Rule 3: the geometric description of a super-entity type is deduced from the geometry of its sub-entity types. For example a super-entity type, such as an hydrographic network, grouping linear (rivers) and polygonal (lakes) elements can be a complex network but not a point. Similarly, a super-entity type grouping only polygonal elements can be a polygon but not a line.

Rule 3.1: if all sub-entity types are mapped and have identical geometrical descriptions, then the super-entity type inherits this geometric description and becomes the only one needing to show the SMS symbol (all sub-entity types sharing this symbol with the super-entity type). For example, the Forest Operation entity-type would show the polygonal symbol, but the sub-entities Spraying, Cutting, Soil preparation, Thinning out, and Planting would show no SMS symbol. On the other hand, if the sub-entity types have different geometrical descriptions, they all keep their SMS symbol and the super-entity type shows its own (e.g. Hydrographic Network, River and Lake entity types all show their own SMS symbol).

ZONE D'EVENEMENT																			
<p>numéro date superficie</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p style="text-align: center;">PERTURBATION NATURELLE</p> <p>mortalité</p> <table style="width: 100%; border: none;"> <tr> <td style="border: 1px solid black; padding: 2px; text-align: center;">FFU</td> <td style="border: 1px solid black; padding: 2px; text-align: center;">CHABLIS</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px; text-align: center;">EPIDEMIE</td> <td style="border: 1px solid black; padding: 2px; text-align: center;">VERGLAS</td> </tr> <tr> <td colspan="2" style="border: 1px solid black; padding: 2px; text-align: center;"> <p style="text-align: center;">MALADIE</p> <p style="text-align: center;">nom de de maladie</p> </td> </tr> </table> </div>	FFU	CHABLIS	EPIDEMIE	VERGLAS	<p style="text-align: center;">MALADIE</p> <p style="text-align: center;">nom de de maladie</p>		<div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p style="text-align: center;">INTERVENTION FORESTIERE</p> <p>commentaire</p> <table style="width: 100%; border: none;"> <tr> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">APPLICATION</p> <p>méthode d'application produit utilisé coût total</p> </td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">PLANTATION</p> <p>racine type de plantation essence des plants nombre de plants</p> </td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">COUPE</p> <p>sorte de coupe type de coupe</p> </td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">INSECTICIDE</p> <p>objectif état final coût moyen à l'hectare</p> </td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">ECLAIRCIE</p> <p>type d'éclaircie</p> </td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">DEGAGEMENT</p> <p>type de degagement</p> </td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">FERTILISANT</p> </td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">PREPARATION DE TERRAIN</p> <p>type de préparation</p> </td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">NETTOIEMENT</p> <p>méthode</p> </td> </tr> <tr> <td></td> <td></td> <td style="border: 1px solid black; padding: 2px;"> <p style="text-align: center;">DEPRESSAGE</p> </td> </tr> </table> </div>	<p style="text-align: center;">APPLICATION</p> <p>méthode d'application produit utilisé coût total</p>	<p style="text-align: center;">PLANTATION</p> <p>racine type de plantation essence des plants nombre de plants</p>	<p style="text-align: center;">COUPE</p> <p>sorte de coupe type de coupe</p>	<p style="text-align: center;">INSECTICIDE</p> <p>objectif état final coût moyen à l'hectare</p>	<p style="text-align: center;">ECLAIRCIE</p> <p>type d'éclaircie</p>	<p style="text-align: center;">DEGAGEMENT</p> <p>type de degagement</p>	<p style="text-align: center;">FERTILISANT</p>	<p style="text-align: center;">PREPARATION DE TERRAIN</p> <p>type de préparation</p>	<p style="text-align: center;">NETTOIEMENT</p> <p>méthode</p>			<p style="text-align: center;">DEPRESSAGE</p>
FFU	CHABLIS																		
EPIDEMIE	VERGLAS																		
<p style="text-align: center;">MALADIE</p> <p style="text-align: center;">nom de de maladie</p>																			
<p style="text-align: center;">APPLICATION</p> <p>méthode d'application produit utilisé coût total</p>	<p style="text-align: center;">PLANTATION</p> <p>racine type de plantation essence des plants nombre de plants</p>	<p style="text-align: center;">COUPE</p> <p>sorte de coupe type de coupe</p>																	
<p style="text-align: center;">INSECTICIDE</p> <p>objectif état final coût moyen à l'hectare</p>	<p style="text-align: center;">ECLAIRCIE</p> <p>type d'éclaircie</p>	<p style="text-align: center;">DEGAGEMENT</p> <p>type de degagement</p>																	
<p style="text-align: center;">FERTILISANT</p>	<p style="text-align: center;">PREPARATION DE TERRAIN</p> <p>type de préparation</p>	<p style="text-align: center;">NETTOIEMENT</p> <p>méthode</p>																	
		<p style="text-align: center;">DEPRESSAGE</p>																	

Figure 5: example of generalisation.

Generalisation Rule 4: a super-entity type inherits the database symbol only if all its sub-entity types have one; then, the sub-entity types do not need to show this symbol anymore (because they now share it with the super-entity type). If not all sub-entity types have the database symbol, the ones having this symbol keep it and the super-entity type shows no such symbol.

Generalisation Rule 5: super-entity types can also be generalised.

Applying these rules to the previously mentioned forest example has led to a reduction of 74 attributes and 36 relations for an addition of 5 super-entities.

CONCLUSION

A good formalism at the *conceptual level* must be simple to understand, have a strong expressive power, and be as rigorous as possible. The development of the three extensions presented in this paper aims to meet this goal. For example, the modelisation of a medium size database for a forestry application has allowed us to include *cartographic only* objects in the conceptual data model and to reduce its size by 50% while improving its readability.

Such results have shown to be useful and practical. However, this still is in development and additional research issues remain for SIS data modeling: modeling spatial relationships, generalizing relationships, dealing with multiple geometric descriptions of entities (e.g. at different scales), building rigorous rules to translate the conceptual schema into the logical and internal schemas, extending the SMS technique to 3-D, extending the data dictionary, applying the SMS technique to management of time-related entity types and relationships, and dealing with mutual exclusivity and mutual inclusivity of relationships.

REFERENCES

ANSI/X3/SPARC Study Group on Data Base Management Systems 1975, Interim Report, FDT (ACM SIGMOD Bulletin) 7, No. 2.

Bédard Y. 1988, On Spatial Data Modeling (First Draft). Third Internal Seminar on Trends and Concerns of Spatial Sciences, Dept. of Geodetic Sciences and Remote Sensing, Laval University, Québec City, June 6-8.

Blais R. 1987, Theoretical Considerations for Land Information Systems: Canadian Surveyor, Vol.41, No.1, pp. 51-64.

Boutin G. 1988, Etude de l'applicabilité d'une méthode traditionnelle de conception de système d'information dans le contexte d'un système d'information à référence spatiale: M.Sc. Thesis, Dept. of Geodetic Sciences and Remote Sensing, Laval University, Québec City, 206 p.

Brodie, M.L. 1984, On the Development of Data Models: in Brodie M.L., J. Mylopoulos and J.W. Schmidt, On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases, and Programming Languages: Springer-Verlag, New-York, pp. 19-47.

Chen P. 1976, The Entity-Relationship model: Toward a Unified View of Data: ACM Transactions on Databases Systems, 1,1, pp. 3-36.

Codd E.F. 1972, Further Normalization of the Data Base Relational Model: Prentice-Hall, Data Base Systems, Courant Computer Science Symposia Series, V.6.

Collongues A., J. Hugues and B. Larouche 1986, MERISE méthode de conception: Dunod Informatique Ed., 211 p.

Date C.J. 1986, An Introduction to Database Systems, V.1, 4th Ed.: Addison-Wesley, 639 p.

Egenhofer M. 1987, Appropriate Conceptual Database Schema Designs for Two-Dimensional Spatial Data Structures: Technical Papers, ASPRS-ACSM Annual Convention, V.5, Baltimore, March 29-April 3, pp. 167-179.

Frank A. 1985, Class notes: Surveying Eng. Dept., Univ. of Maine, Orono, USA.

Martin J. and C. McClure 1988, Structured Techniques: the Basis for CASE, Revised Edition: Prentice-Hall, 776 p.

Paquette F. 1988, Utilisation et adaptation du formalisme entité-relation pour la structuration d'une base de données à référence spatiale avec application à la gestion des données forestières. Unpublished paper presented at the Dept. of Geodetic Sciences and Remote Sensing, Laval University, Québec City, Dec. 7.

Shlaer S. and S.J. Mellor 1988, Object-Oriented Systems Analysis: Modeling the World in Data: Yourdon Press (Prentice-Hall), 144 p.

Tabourier Y. 1986, De l'autre côté de MERISE: Editions d'Organisation, 241 p.

Tardieu H., A. Rochfeld and R. Colletti 1986, Méthode MERISE, principes et outils, Tome 1, 2nd ed.: Editions d'Organisation.

Ullman J.D. 1982, Principles of Database Systems: Computer Science Press, 484 p.

Wang F. and Newkirk R. 1988, An Entity-Relationship Model for Geographical Information System Development: ACSM-ASPRS Annual Symposium, Saint-Louis, V.5, p. 162-172.

This research has been financed by grants #1264 and #A5742 of the Natural Sciences and Engineering Research Council of Canada.

A Fully Integrated Geographic Information System
Dr. John R. Herring
INTERGRAPH Corporation
One Madison Industrial Park
Huntsville, Alabama 35807-2180

ABSTRACT

The major problem in GIS implementation is the diversity of data types and data base management systems that can carry geographically related data. A fully integrated GIS must be able to handle data in various formats (vector, topology, attribute, raster, grids, TINs, etc.), providing a environment where they can co-exist, and interact. It must also provide standard interfaces to external data bases; foreign in structure, schema, DBMS, and machine environment. This paper describes the some requirements for and approaches to a fully integrated GIS.

INTRODUCTION

The last thing the Geographic Information System (GIS) community needs is yet another definition for GIS. Some current definitions are "that which exists" (a current useful product in geographic analysis) or "that which I do" (based on a particular product or application). This paper tries to remove this bias and establish a definition and basic requirements for "that which should be."

To begin, let us distinguish between data management and applications. The need in the GIS user community is for applications, which perform tasks for data collection, extraction, analysis, and product generation. But, neither singly nor collectively, do applications support the data management tasks most effectively done by a DBMS, usually a Relational Data Base (RDB), in the business community. If we make the same distinction, defining a GIS as "a DBMS for data having geographic significance", or "a DBMS upon which geographic applications can be built"; then a GIS must:

- o provide a common interface to organize, load, extract, and report on all types of geographic data.
- o provide a generic query environment to perform analysis common to most applications.
- o maintain ("serve, protect, and defend") the integrity and the validity of geographic data

The rest of this paper draws conclusions about what such a GIS must be, and presents requirements and potential solutions to fundamental problems. Since no restatement can truly begin unencumbered by the past, several technical controversies on GIS implementation are addressed directly.

WHO USES A GIS

A GIS manages geographically significant data, and supplies combined spatial and attribute analysis tools. Given this, who are the users of a GIS? We can classify the handling of geographic data into four categories (not necessarily mutually exclusive): collect, merge, validate, analyze, and produce products.

To collect data is to extract it from non-data-base sources, verify its consistency with that source and convert it to the appropriate data base format. The immensity and the importance of the verification task requires a number of automated and semi-automated tools to discover potential errors before the source is released. This includes such diverse problems as attribute range, consistency and validity checking; and geometric anomaly prevention, detection and correction. These tools require a GIS. In fact, the verity of the collected data is so pivotal to the validity of the results of any GIS application, and analysis is so important to verification, that it should be emphasized that no one needs GIS functionality more than the person collecting data.

To merge data is to combine multiple sources of collected data into a single environment appropriate for further manipulation. The merge process must be able to recognize when two feature representations are in fact different versions of the same feature, separated by collection method, statistically reasonable error, or time; and to be able to combine these multiple representations in a statistically valid compromise. A GIS is needed.

To validate data is to assure its internal consistency. To analyze data is to derive implicit information from explicit data. Combined spatial-attribute validation and analysis are the classic GIS applications, often confused with the GIS itself. They certainly require a GIS.

To produce products from data is to convert the data base information into a form directly consumable by a client (either a human or other digital process). As such, it can involve a massive reshaping of the data. For example, the production of paper or digital maps must contend with generalization, agglomeration, aggregation, and conflict detection and resolution. Even in a semi-automated environment, this very complex spatial analysis requires a GIS.

So the answer is "anyone who handles geographic data."

DATA TYPE COEXISTENCE

The first major separation between GIS's and other DBMS's, is the data itself. No non-spatial, and few non-geographic data bases have such a wide variety of data types. These types include, but are not limited to the following:

- o feature attribute information;
- o structured (topological) vector graphics;
- o unstructured vector graphics;
- o raster representations maps, aerial and satellite photography (grey-tone, color and multi-spectral);
- o TINS (triangulated irregular networks), grids, contours and other elevation models;
- o non-elevation TINS, grids, and raster information for the representation of analytical surfaces (e.g. soil permeability, cost functions, demographics);
- o 3 and higher dimensional equivalents for TINS (simplicial complexes) for subsurface geology;
- o projection, transformation and other coordinate information (including projection parameters, datum; primary and alternate units of measures; digitizer setups (table to screen); co-registration parameters

- (e.g. best fit functions between raster-vector, raster-raster, or vector-vector data);
- o relational information linking all of the other data types together;
- o schema information, describing the application specific parameters for the other types of data (attribute names, types, and ranges; feature types, representation rules and display parameters; etc.).

The first coexistence problem is to co-register any of the various data types to common geographic coordinate frames of reference; but this is only the beginning. The disparate data types must be simultaneously manipulable; all data from a single geographic area, regardless of format or content, must be read and write accessible within a single GIS process; subprocesses such as commands must manipulate as many of the data types as logically feasible.

Structured Versus Non-structured

There is a continuing controversy in the GIS community between two seemingly disparate philosophies; between "non-ambiguous structured data model" and "a flexible and simplified data model." To explain this, let us investigate some of the major points of contention.

Real-time topology versus on demand structuring: No one disagrees that topologically structuring of vector data is a boon to spatial analysis (Herring-86), but there is a controversy as to the manner in which the system maintains such a structure. The two logical alternatives are real-time maintenance (topology is always valid) and on-demand updating (topology is frozen at user chosen points and bulk updated or recalculated as necessary for analysis). These two alternatives are solutions to different problems. In older and less powerful workstations or systems, the real-time load for topological maintenance quickly uses unacceptable amounts of system resources. In the newer workstations or system (with 0.5 MIPS or better per user), the excess capacity of the system can be used to take advantage of a real-time maintenance system. These advantages include at least the following:

- o ad-hoc combined spatial-attribute query and analysis
- o real-time geometric anomaly detection and prevention
- o better system utilization (% use of machine cycles)
- o intelligent user-feedback based on automatic analysis

The most obvious recipients of these benefits are the data collection tasks (with the real time validation and verification) and what-if-analysis tasks (faster turn-around on spatial queries or analysis based upon recent and tentative geometric changes). In summary, real-time systems provide a more flexible environment, and are preferable assuming sufficient system resources to maintain interactive response times. In non-interactive processes, maintenance is always less expensive than reconstruction.

Relational versus Object-Oriented Data Model: This is a true non sequitur for a very simple reason: almost all GIS's (even those using a RDB) are already object oriented (see Ullman-88). An "object-oriented system" is one which supports an abstract concept "object" ("entity", "feature") having existence independent of any attributes that entity may or may not have. The opposite of this is a "value-

oriented system" which models only attributes. For example, in a valued-oriented system such as a "classical" RDB, the entity (and any tuple representing it) owns its existence to a non-null key attribute combination, and any pair of entities (sets of tuples) having equal (key) attributes are equal (duplicates are eliminated). Such pure value-oriented systems can create confusion, such as two employees with the same name getting each others paychecks. To prevent this, most applications using RDB's assign "employee numbers." This same natural approach appears in linked graphics-attribute systems, where graphics are given identifiers ("graphic link", or "feature id"). This moves the GIS from the "value-oriented" world to the "object-oriented" world, since the system now gives meaning and existence to the graphic or feature "objects" independent of their spatial and non-spatial attributes. This does not mean that Object Oriented Data Bases (OODB's) are being used, since other requirements are levied against such system (at least encapsulation and abstraction), but it does imply that, taking use into consideration, many of the object-oriented concepts are natural extensions of the relational model. In fact, some classical problems in RDB design, referential integrity and normal forms, lead into object concepts. For example as early as 1980, three rules for converting an entity-relation model to a relational one were laid down by Wong and Katz (Chap. 21, Stonebreaker-86) ("() added to identify equivalent OODB concept):

- 1) each entity set (object) has an explicit identifier (object id) which represents it globally in the relational model
- 2) the identifiers (object id's) of a primitive object (class) together with all the primary functions (attributes) of the primitive object are grouped in the same relation in the relational schema
- 3) there is one and only one primitive object (class) per relation of the relational schema

They go on to show that these "mapping rules" lead to fourth normal form (4NF) RDB implementations. Looking back on this from the OODB point of view, we see that Wong and Katz have essentially proven that a straight forward, formal implementation of an object model in a RDB gives a 4NF relational data base schema. Since this early parallelism, further work has brought the two data models closer together. For example, Rumbaugh-87 proposes including relations in formal OODB models; Blaha-88 suggest using object-oriented models to design RDB schemas, formalizing what now occurs naturally (see above); and Ullman-88 proposes an extended entity-relation model as a super-model for both RDB's and OODB's.

A complete GIS system, regardless of its implementation details, must be able to communicate with all data bases capable of storing geographic data, including both RDB's and OODB's. Since DBMS can only rarely exchange raw data, this implies compliance with standard exchange formats and protocols (possibly based on object-oriented extensions of SQL standards, see Herring-88).

Single-content Multiple Layers versus Multiple-content Single Layer: (whether to merge all the data into a single integrated geometric structure or to maintain separate layers, merging only for analysis). These two options are

actually the opposite ends of a broad spectrum of possible data-base schemes involving various levels of integration. Each application, and each user has differing needs based upon the particular workflow, analysis requirements, processing power, inter-department interfaces, etc. Even then, the requirements may vary between subsystems within the users' DP community. The considerations in deciding which data types (themes) to aggregate into single topologies (layers) are quite varied. They include at least the following:

- o the amount of shared geometry between the themes,
- o the degree in which they are combined in the usual course of GIS processing,
- o the common source and maintenance responsibility.
- o data complexity and storage requirements within a theme, and
- o the geometric or non-geometric quality of a theme.

Sliver and gap detection, elimination and prevention is very costly, so that themes which tend to share a great deal of their geometry should probably be integrated at the master data base level to prevent high initial access time for applications requiring integrated data. The same is true for data that is often used in combination during GIS processing, again to save the repeated cost of the merge process. Common source and maintenance responsibility allows for easier integration.

So each user must be able to select from the broad spectrum of data structures, most often deciding on a master data base with multiple layers each with multiple content.

Implicit Data versus Explicit Data: In a RDB, some contend that relationships are only implicit since they are "derived" through the join process. While technically correct, this view ignores the fundamental problem of referential integrity. Suppose for example, that in a Wong-Katz RDB implementation, there are tables for road "segments" and other tables for "highways," and that these entities are linked via a specific ownership relation. Thus, one of the segment tables would have an attribute for highway-number, acting as a foreign key into the highways tables' primary key. If the user wishes to report on "all road segments that belong to particular highways," then the needed join on highway-number is valid since a common value means ownership (is valid semantics). If the data base exhibits referential integrity, then each value of highway-number in a segment table corresponds to a valid and correct highway-number in a highway table. On the other hand, if the join were done on other integer valued fields (such as segment.width-in-meters = highway.age-in-years) the results might be nonsensical. Thus, even though the tables appear to be without structure, there is an implied structure based upon the semantic meanings of attributes.

The distinction then between structured and unstructured data is whether or not the DBMS system is aware of the structure that naturally exists within the data. In a GIS, this includes common attributes and common geometry. In a RDB environment, this means that the system is aware of any foreign key attribute value and support procedures to maintain referential integrity; in a OODB, this means that any foreign key is implemented through object relations.

Valid non-causal joins (those based upon attribute relations not involved in foreign-key to primary-key linkage), represent a extra-system (user) interpretation of the data. Thus, the user is placing an interpretation on the data base not considered during its data structure design phase. Whether this query derives valid information depends upon the correlation between the semantic interpretation of the data analyst and the data extractor.

Thus, a GIS, during normal operations, should maintain the correct interpretation of the data (referential integrity). But, in non-causal joins such as used in statistical analysis, the GIS must allow the user to override the default interpretations.

Integrated versus Linked Attribute-Graphics

The most obvious problem in a system supporting such a wide variety of data types is to minimize the number of lower-level data management systems involved in the GIS. The GIS must hold and manipulate several types of stored (persistent) data as listed above, in addition to non-persistent, application data (point buffers, the temporary results of query, display information, window information, temporary command information, etc.).

The most controversial element is the linkage between vector (or topological) data and feature attribute data. The classical approach is to split the two data types at the graphics-to-attribute juncture. This creates a gap that must be hurdled every time a combined spatial-attribute edit or query is executed. The extreme alternative is to place all attribute information in the same DBMS as the graphic information. This is not generally feasible in the RDB world due to its value-oriented programming implementation. Graphic operations (such as simple display) require the access of large amounts of diverse data (an average graphic window might contain 500 kilobytes of information distributed among 2,500 data items of differing types). This would require an RDB to access multiple tables, and return large amounts of structured data. Currently, commercial RDB's are simply incapable, by several orders of magnitude even on the best systems, of reasonable response times in such situations. Further complicating the process, RDB's usually require 1NF (first normal form, all column data types are simple), while geometric data is inherently not 1NF (e.g. coordinate lists). OODB's by their very definition, and some of the more theoretical RDB's, support abstract data types and alternate access methods such as triggers (Stonebreaker-86, Andrews-87) which render such problems solvable.

On the other hand, a single data management system for attributes and geometry simplifies the system, making real-time topological maintenance (see above) and integrated spatial-attribute ad-hoc query (see below) possible. This implies, as in the real-time topology discussion, the implementation of the geometry-attribute linkage is a function of the performance level and sophistication of the system.

When large amounts of static, persistent data are linked with the geographic data (such as well logs in petroleum applications), data size can become a problem. Therefore,

the application or user must decide which data to integrate and which to segregate (to other nodes in a distributed system, see below). The criteria that affect this decision are basically the same as the layering criteria below (i.e. some of the layers are non-geometric). Some data such as raster, and dense grids are so storage space intensive, they probably should be stored in the most efficient manner possible such as indexed run-length-encoded or quadtree structured files. Even so, foreign linkages must exist in main DBMS to hold integration information such as co-registration parameters (see below).

In summary, some degree of tight integration between attributes and geometry is required to support full GIS capabilities; but, the degree of integration between layers of the data should be user and application controllable system configuration decisions.

Communication

Once coexistence has been achieved, the next most important criteria is communication between the data types. For example, the passing of geometric descriptions should be possible between any two domains. For example, the raster subsystems must be able to perform classification algorithms based upon vector area criteria; and raster line-following and polygon classification must be usable in vector digitization and spatial query. This requires that each of the data subsystems support common protocols for the transfer of at least geometric information. Across diverse systems this implies universally acceptable exchange standards. Within a single multi-content system, internally defined protocols are more efficient.

SPATIAL QUERY

It is in spatial query that a GIS is most distinct from standard DBMS implementations. For example, in a RDB there is no interpretation of the data in spatial terms, since such interpretations depend upon the particular abstract representation of space chosen by the application. Thus no reasonable set of spatial operators can exist in the pure form of the RDB model. On the other hand, a GIS data base system sole purpose is to incorporate spatial operations into the other more conventional DBMS functions.

This leads to a nearly insolvable problem. Spatial extent is different from other data types: non-1NF, non-declarative geometric algorithms even for simple comparison (e.g. point-in-polygon), etc. In fact, many spatial data bases separate the spatial and non-spatial data into two systems (see above). This usually means that an integrated spatial query language is impossible, and such systems must rely on a tool-box approach to spatial analysis, implementing spatial operators as separate procedural code which the user alternates with the non-spatial query language to do analysis. This is the ultimate (worst) in procedural approaches, forcing the user to specify each transition from the spatial arena to the non-spatial arena ("procedural" queries specify how data is manipulated, "non-procedural" or "declarative" query specify what results are required, leaving the procedural decisions to

the DBMS). Spatial operators are thus object-oriented and essentially procedural, and out of the usual domain of a declarative query language.

This leads to an interesting paradox. GIS's built on RDB's should expect to gain the advantage of the non-procedural, declarative query language, but do not due to the nature of the spatial data; leading to a procedural spatial query and analysis environment. GIS's built on OODB's, which are naturally procedural (see Ullman-88), supply as applications those procedures needed to do spatial query and analysis, thus allowing the GIS users and applications to reside in a declarative, non-procedural environment (see Herring-88).

FOREIGN DATA BASE LINKAGES

The data size in GIS's, and the need to access large volumes of non-spatial data, require that the question of distributed, non-homogeneous data bases be addressed. Since much has been written on the general problem of distributed data bases (e.g. Ullman-88), we will concentrate on the mechanism to link the data within the data bases together.

Geographic Linkages: The most common way of linking GIS data is common geographic location. For disparate data contents (different layers), this is sufficient since most co-location is not causal and subject to some statistical interpretation anyway (this data-layer independence is a measure of the success of the layering, see above). For common data content (dependent layers or adjacent data collection cells), it is not sufficient. Absolute error is much larger than the possible relative error, and usually larger than the micro-structure of the geometry (such as real road misalignment at intersections).

Explicit Linkages: In a RDB, explicit linkages, as internal foreign keys, are implemented by a set of common attribute values. In a GIS implemented upon the RDB, these linkages can be logically either value-oriented or object-oriented, depending upon whether the system allows direct application or user access to primary and foreign keys. In a OODB, such linkages are maintained by the DBMS itself through linkages based upon internal object id's. Value-oriented linkages are still possible through common attribute values.. As within a single DBMS, concerns of referential integrity suggest that the GIS maintain object-oriented linkages.

THE DATA SERVER

The environment of the GIS with its multiple data formats, partitioned data, multiple foreign data linkages, places heavy requirements on the management of data at the macroscopic level. This management system, here called a data server, must fulfill at least the following functions:

- o maintain data on the content, accuracy, format, geographic extent, and storage location of all data sets, including versioning;
- o maintain source utilization and production histories;
- o translate standard exchange formats to and from internal formats;

- o control access to data;
- o manage schema and view information

Much of these are standard DBMS concerns, so we shall limit ourselves to items which take on a special meaning due to the geographic nature of the data.

The Geographic Index: The geographic index necessary for a distributed GIS is a GIS itself, since the partitions are associated to their spatial extent. Spatial query and analysis tools used directly on the data are needed in the index to support production management and distributed analysis (analysis covering some number of partitions, executable from the index, which distributes processing and data to the appropriate partitions, possibly distributed).

Schema Management: In any non-homogeneous distributed data base, the schemas of partitions may vary based on layer, data format and type. Further, data collection and various analysis tasks may require different schemas (simple for collection, complex for analysis), which in turn differ from master data base schemas. The data server must provide a set of schema management tools such as:

- o a generic schema definition interface independent of underlying DBMS's (facilitating inter-DBMS exchange);
- o the association of partitions to appropriate schemas;
- o update functions for maintaining consistency between the schema and all partitions of a single layer;
- o a schema merge capability to allow multiple layers to be combined into single data sets for analysis;
- o translation functions between different schemas and user views to allow inter-application sharing of data

This schema management requirement is independent of whether the GIS is based upon a RDB or an OODB. Currently, most non-geographic systems avoid this problem by insisting upon "all data in a single data base," an unacceptable approach for GIS's due to data volume and diversity.

Access Control Locking and Concurrency: Access control requirements in a GIS differ widely from those in conventional DBMS's. First, standard record, or table locking is nearly useless, since data is usually accessed based upon common location, not data type. Some form of "area locking" is more appropriate. This is especially true for topologically structured layers, where a few data types (face (polygon), edge (arc) and node (point)) are evenly distributed so that a single table lock could bring all but one user to a halt. This is even a greater problem in a RDB which locks tables after a threshold number of tuples have been locked. There are two basic solutions: partitions locking or object-oriented locking. In the first, a partition is locked completely while a user modifies it, resulting in a long transaction (hours or days). In the second, side effect locks can be controlled at the object class level. In a topological data base (Herring-87), standard locks could be used on feature data, and proximity locks on topology (not allowing two users to modify the same or adjacent faces simultaneously).

SUMMARY

The GIS implementations problems are much more complex than found in any non-spatial DBMS. Thus, not only should GIS research investigate the forefront of data base technology,

it should also be driving RDB and OODB research to investigate specific geographically-related problems. Further, we must recognize the inevitability of diverse GIS data and GIS implementations and concentrate on a rational set of exchange standards capable of supporting diverse, real-time, distributed, geographic processing.

ACKNOWLEDGEMENT

The issues and solutions presented in this paper represent six years of continuous interaction, within Intergraph and the GIS community. I wish to thank everyone who has participated in any of these discussions, especially those directly involved with TIGRIS requirements analysis, design, and implementation.

REFERENCES

- Andrews, Timothy, and Craig Harris; "Combining Language and Database Advances in an Object-Oriented Development Environment"; OOPSLA'87 Proceedings; ACM; October 4-7, 1987; pp 430-440.
- Blaah, Michael R., William J. Premerlani, and James E. Rumbaugh; "Relational Database Design Using an Object Oriented Methodology"; Communications of the ACM; vol 31: no 4; ACM; April 1988; pp 414-427.
- Herring, John R., "TIGRIS: Topologically Integrated Geographic Information System"; Proceedings of AutoCarto8, March 1987, Baltimore, Maryland, pp. 282-291.
- Herring, John R., Robert C. Larsen, and Jagadisan Shivakumar; "Extensions of the SQL Query Language to Support Spatial Analysis in a Topological Data Base"; GIS/LIS'88 Proceedings; ACSM, ASP/RS, AAG, URISA; November 30 - December 2, 1988; pp 741- 750.
- Rumbaugh, James; "Relations as Semantic Constructs in an Object-Oriented Language", OOPSLA'87 Proceedings; ACM; October 4-7, 1987; pp 466-481.
- Stonebreaker, M. 1986, ed. The INGRES Papers: Anatomy of a Relational Database System, 1986, Addison-Wesley, Reading, Massachusetts.
- Ullman, Jeffery D.; Principles of Database and Knowledge-Base Systems, Vol 1; Computer Science Press; Rockville, Maryland; 1988.

SPATIAL TOOLS FOR THE ADMINISTRATION OF
MAJOR INSTITUTIONS

Jeffrey M. Young
Program Administration Group
16511 Martha Street
Omaha, Nebraska 68130

BIOGRAPHICAL SKETCH

Jeffrey M. Young was born in Scranton, Pennsylvania in 1954. Mr. Young attended public schools in Scranton and graduated *cum laude* from Lock Haven State College receiving a baccalaureate degree in Geography. Mr. Young continued his education at Arizona State University where he was granted a Master of Arts degree in Geography. While in pursuit of the Master's degree, he participated in the National Science Foundation supported program, "Spatial Analysis of Land Use." Mr. Young is currently a Senior Consultant with Program Administration Group.

ABSTRACT

Administrators of major institutions are seeking new tools to aid in the management of geographically dispersed facilities. Typically these centrally-controlled institutions are a collection of several semi-autonomous units such as state prisons, hospitals, and universities. Traditional information processing approaches for these institutions have relied upon Management Information System (MIS) methodologies. Improved spatial information processing tools provide an opportunity for institutional planners, operators, and maintenance specialists to migrate from a non-graphic MIS environment to a spatially-oriented setting. Both Geographic Information Systems (GIS) and Computer-aided Drafting Systems (CAD) have significant roles in this transition. A model conceptual design of a spatially-related Institutional Information System (IIS) is presented in this discussion. The design is multi-scaled to accommodate the requirements of an institution as a whole, as well as site and building details, for routine operation and maintenance at each location. The IIS conceptual design is structured to support the life cycle of the institutions, i.e. planning, design, construction, and operation and maintenance; including pre-programming, space planning, master planning, resource allocation, staffing, cost analysis, remodeling, rehabilitation, and inventory control.

OVERVIEW

Technology-and-growth enthusiasts would like us to believe that technology and capital can solve almost all problems (Kahn, 1976). The author of this paper does not embrace this extreme perspective. However, the paper is prepared with a spirit of guarded optimism; institutions, such as universities, medical facilities and prison systems, need to prepare for the 1990's and can benefit from expanded use of information technology.

In the context of this discussion, an institution is any organization established to conduct the business and/or operations of a society or association. These institutions may be private in character; designed for profit or publically supported; being operated for the well being of a constituency. Institutions rely heavily upon estimates, projections, and forecasts to evaluate facility conditions and function as a part of normal operation. Institutions are the fabric of our nation. These include schools, universities, banks, religions, insurance companies, health care facilities, prisons, the military, airports, and cultural and historic centers to name a few. We all benefit by well run institutions and, conversely, we all feel the impact of institutions under stress. All institutions have limited staff, funding, and space resources; and some are confined to a cramped collection of buildings, people, and cars with little room for growth. Under these conditions prudent allocations of staff, budgets, and space is of primary concern along with maintenance of existing buildings, grounds, and infrastructure. As a group, institutions are used for diverse functions, however, most have been built from scratch, are long lived, and are surrounded by ever changing land uses and landscapes (Lynch, 1971).

Quality information is required to efficiently plan, design, construct, operate, and maintain an institution. Inadequate estimates or projections have contributed to the failure of institutions (Hall, 1980). The value of map data for facility siting is well established (Williams, 1983). Certainly engineering and architectural drawings and specifications are a prerequisite for design and construction. It is a pity that for most institutions the information gathered during planning, design, and construction has not been effectively integrated into operations and maintenance. Perhaps integration may be too ambitious, but some form of data linkage is appropriate. All too many times a facility manager finds it difficult to answer simple questions such as:

- What is the condition of our buildings, structures, and infrastructure?
- What is the total square feet of our institution?
- What is the total value of our institution?
- How can our functional use of space be improved?
- Where can we build and expand?
- What needs to be repaired, renovated, or decommissioned?
- Have these repairs been completed and, if not, when will they be done?
- What are our operation and maintenance costs next year? ...the next five years?

Over time, a facility manager can find himself responding to a series of ad hoc inquiries rather than attending to daily needs of the facilities. He may encounter islands of automation in his search for an answer, but in the end some degree of uncertainty and temporal error is present in his response to the questions listed above. The data required to answer those questions may exist, but not in a form for his purposes. Improved information management is now mandated. Long-term institutional data managers, who serve the needs of facility managers and institutional planners, must develop information systems with several attributes including (after Zimmerman, 1987):

- Large storage capacity with minimum operator intervention required
- Accessibility to a wide range of users
- Flexible archiving and networking
- Automated data management
- Responsiveness to long-term growth requirements and technological improvements
- Security

Computer technology to support all aspects of the life cycle of facilities has been improved and refined to a point where implementing an IIS is practical. Presently most institutions possess a disjointed collection of data, procedures, and computer hardware and software which, when approached by facility managers and institutional planners, has been a source of frustration. Database and intelligent graphic-oriented tools may ease the frustration of these users.

This paper presents a definition of an IIS, provides a model IIS conceptual design, and describes IIS implementation steps.

INSTITUTIONAL INFORMATION SYSTEMS

Institutional Information Systems (IIS) comprehensively provide for the collection, data preparation, storage, management retrieval, analysis, synthesis, and display of data on the institution as a whole; campus sites and surroundings; structures and buildings on campuses; building systems; and equipment within each building and structure. An IIS is an organized collection of data, procedures, personnel, computers, software, and communications. Views of the data within an IIS can be tabular, graphic or both. An IIS provides institutional planners and facility managers with a means to receive, sort, retrieve, and transmit information. Information can include region-wide displays with associated data covering several states or detailed inventories of fixed assets within a particular building. Often data on an institution is stored in a variety of media at numerous locations. IIS's can be developed to reduce data lost and unnecessary duplication.

In effect, an IIS is a super-information system which is intended to link or loosely couple several spatial and non-spatial, and, graphic and tabular data handling subsystems which include:

- Administration including staffing, financial, inventory control, and purchasing functions (MIS-based)
- Asset database (MIS-based with a CAD/GIS link)
- Facility database (CAD/GIS-based)
- Infrastructure database (CAD/GIS-based)
- Maintenance management (tabular with a CAD link)
- Environmental compliance (GIS-based)
- Public affairs and relations
- Real estate acquisition and disposal
- Architectural/engineering planning and design (CAD-based)
- New construction and renovation (CAD-based)
- Archiving (tabular)
- Special purpose subsystems (such as statistical mapping and analysis, environmental monitoring, demographic analysis, capital improvement planning, space planning, security, economic forecasting, historical preservation, and litigation support)

MODEL IIS CONCEPTUAL DESIGN

The model conceptual design of the IIS is organized into information tiers (see Figure 1). Each tier is comprised of one or more data element groups which include institution features, building systems, and equipment. This model is intended to be suitable for any of the institutions described earlier. Tiers and data element groups for the IIS are as follows:

<u>Tiers</u>	<u>Data Element Groups</u>
Institution-wide	- campuses - other semi-autonomous locations
Campus Sites and Surroundings	- site features - utilities - structures and buildings - other features
Structures and Building Systems	- operations - HVAC - instrumentation

- plumbing
- power
- lighting
- communications
- other systems

Equipment

- operations
- HVAC
- instrumentation
- plumbing
- power
- lighting
- communications
- other equipment

The Institution-wide Tier contains data of greatest value to institution planners and budget specialists. The Campus Sites and Surroundings Tier provides detailed information on each campus and other semi-autonomous units. This data would provide much needed data to the facility managers at each location. The Structures and Building Systems Tier provides detailed floor plans and characteristics regarding major building systems. The Equipment Tier is the most detailed of all of the tiers in terms of both spatial resolution and associated data on individual pieces of equipment. The associated data for equipment would typically include maintenance cycles, performance standards, description identification number, location, model, type, manufacturer, etc.

Example data profiles for the Campuses Group, Site Feature Group, HVAC Systems Group, and HVAC Equipment Group have been prepared to further explain the content and level of detail of each tier (see Figures 2, 3, 4 and 5). Each profile defines the graphic and non-graphic characteristics of the data element group being portrayed including a definition of the data element group, data types included, a representative plan view, and a listing of typical non-graphic associated data.

IMPLEMENTATION STEPS

Implementation of an IIS requires the execution of several activities which would likely be conducted over a period of months and, in some cases, years depending on the size and physical extent of the institution. A pilot should first be conducted prior to institution-wide implementation. A six phase prototypical implementation approach is described below.

Phase 100-IIS Conceptual Design

The identification and evaluation of user requirements and data sources currently in use will form the foundation for an IIS. The tasks to be conducted during this phase are:

- Task 110 Document IIS user requirements
- Task 120 Review IIS data sources
- Task 130 Design a conceptual IIS database

- Task 140 Define conceptual applications modules
- Task 150 Design a conceptual computer configuration
- Task 160 Recommend an IIS organizational framework
- Task 170 Evaluate costs

Phase 200-Computer System Selection and Acquisition

Computer hardware and software will be selected and acquired during this phase. After acquisition of computer components from vendors, an acceptance period will provide the institution with assurances that the selected system will fully support their requirements. This phase will require the execution of six tasks:

- Task 210 Establish functional requirements
- Task 220 Prepare a computer vendor solicitation
- Task 230 Issue solicitation and receive responses
- Task 240 Evaluate responses and select vendor
- Task 250 Acquire and install computer hardware and software
- Task 260 Evaluate performance and accept equipment

Phase 300-Detailed Database Design

The IIS database design will be based upon the physical requirement of the computer configuration selected under Phase 200 and the functional requirements identified under Phase 100. The physical design of the database will describe keys and links to various applications. The tasks for this phase are as follows:

- Task 310 Specify database
- Task 320 Test database design
- Task 330 Finalize database design

Phase 400-Data Conversion

A digital database for the institution will be created during this task. Prior to conversion, data standards will be established to provide guidelines for data input, format and structure; accuracy and precision requirements; and performance schedules. The tasks which will be addressed in this phase are:

- Task 410 Establish data standards
- Task 420 Select conversion vendor
- Task 430 Test data quality and compatibility
- Task 440 Convert data

Phase 500-IIS Operations

The IIS will begin routine operation during this phase. Successful operation will require clearly defined organizational roles and responsibilities. Some staff training is anticipated. This phase contains several tasks:

- Task 510 Define organizational roles and responsibilities
- Task 520 Select staff
- Task 530 Train staff
- Task 540 Operate the IIS
- Task 550 Provide technical support

Phase 600-Applications Development

Applications will be developed which employ the previously designed database and acquired computer configuration. Initial applications development will concentrate on those applications which will serve a wide range of IIS users, although some applications will meet specific user requirements. These applications will include space planning, code compliance, infrastructure management, inventory control, environmental monitoring support systems, capital improvement planning support, and incidence mapping. The phase involves five tasks:

- Task 610 Select candidate applications
- Task 620 Design selected applications
- Task 630 Test applications
- Task 640 Implement applications
- Task 650 Provide technical support

SUMMARY

An IIS is a set of human and capital resources which provides information to institutional planners and managers. Successful implementation requires the cooperation of all levels of an organization. However, please remember, technology alone solves nothing.

REFERENCES

- Hall, P., 1980, Great Planning Disasters, University of California Press, Los Angeles.
- Kahn, H., W. Brown and L. Martel, 1976, The Next 200 Years, A Scenario for America and the World, William Marrow and Company, Inc., New York.
- Lynch, K., 1971, Site Planning (Second Edition), The M.I.T. Press, Cambridge, Massachusetts.
- Williams, E. A., D. H. Blau and H. R. Schaal, 1983, Siting of Major Facilities, McGraw-Hill Book Company, New York.
- Zimmerman, M. B., 1987, "Information Explosion Mandates Planning," Government Computer News, March 27, 1987.

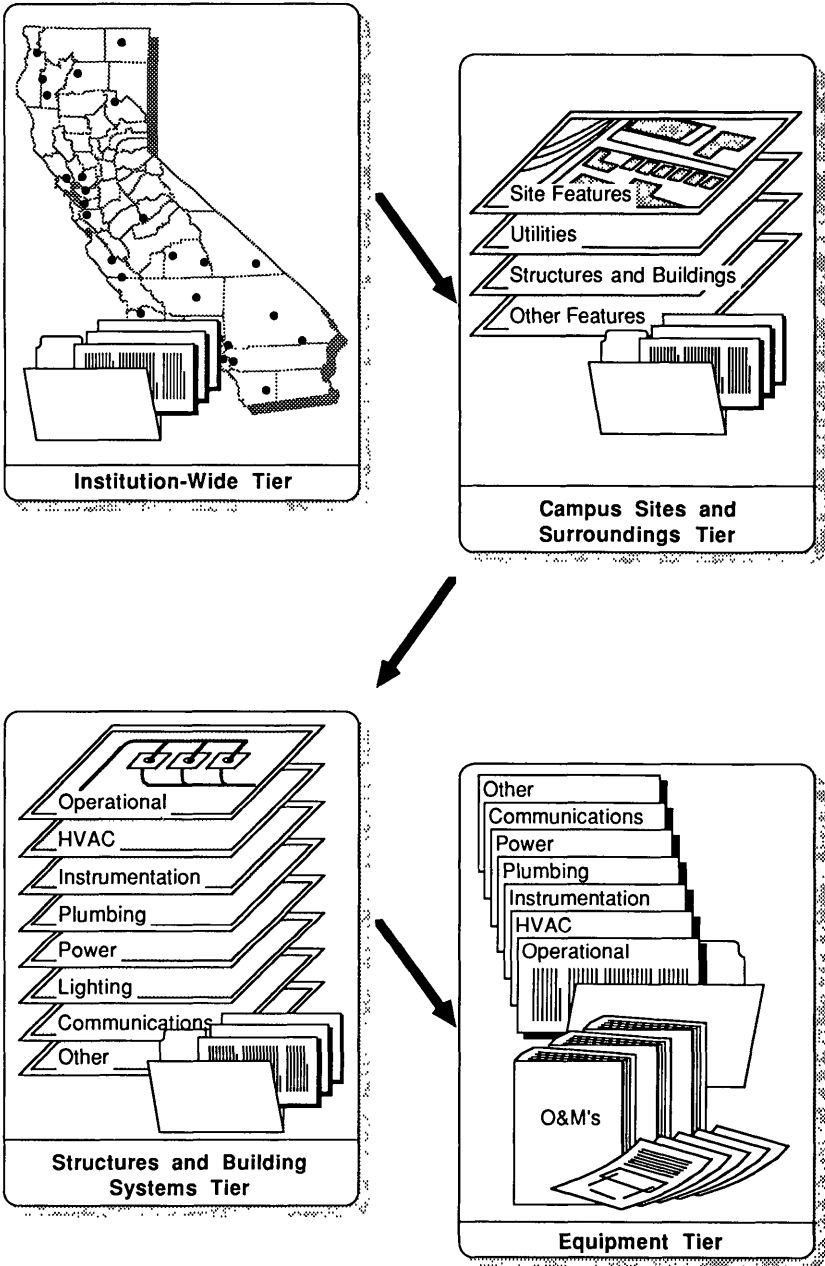


Figure 1. Institutional Information System Model Conceptual Design

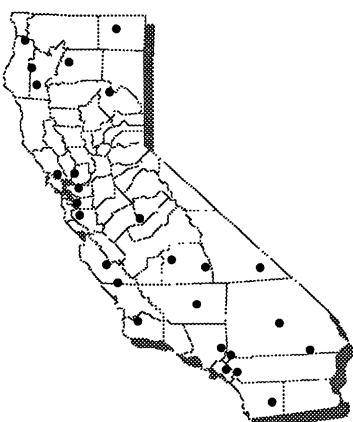
Tier:	Data Element Group:
Institution-wide	Campuses
Definition:	
Campuses: Semi-autonomous contiguous areas with infrastructure, natural features, structures and buildings associated with an institution. Institutions are typically comprised of multiple semi-autonomous campuses.	
Plan View	Associated Data
	<ul style="list-style-type: none"> Functional Uses Year Constructed Total Acres Acres Developed Gross Floor Area by Function Road Length/Area/Media Generalized Utility Information Design Capacity Total Number of Users Total Number of Staff Climate/Meteorology Seismic Zone Surroundings Land Use Other

Figure 2. Campuses Data Profile

Tier:	Data Element Group:
Campus Sites and Surroundings	Site Features
Definition:	
Site Features: Natural and manmade components of a campus site and surroundings including hydrography, pavement, roads, paths, property, boundaries, sidewalks, and vegetation.	
Plan View	Associated Data
	<ul style="list-style-type: none"> Description Location Size Dimensions Materials of Construction Year Constructed Inspection Interval Last Inspection Staff Requirements

Figure 3. Site Features Data Profile

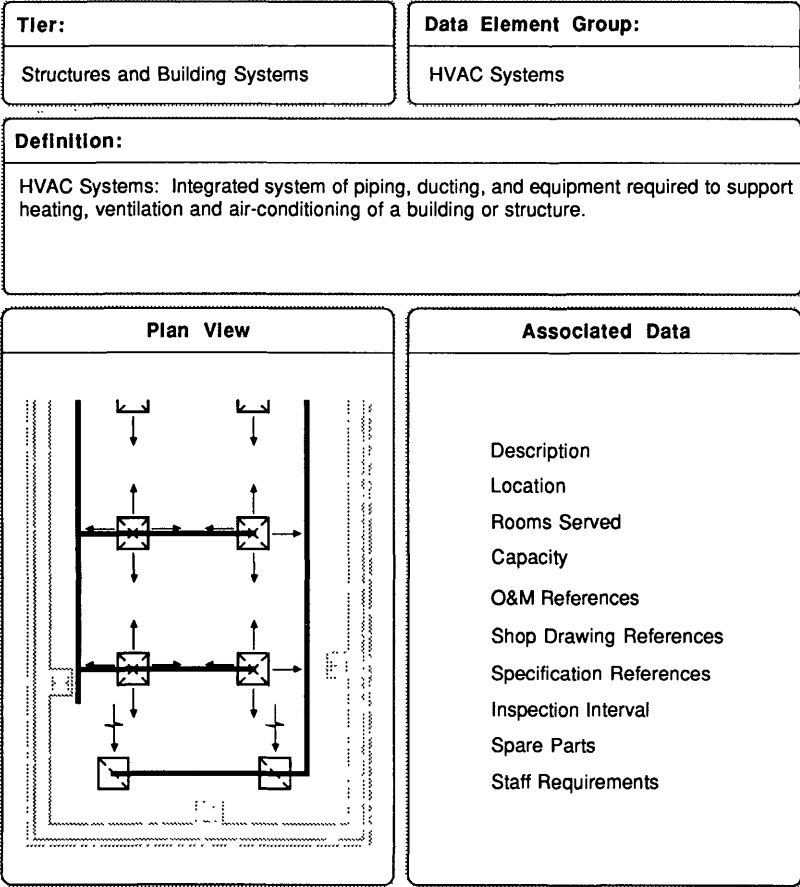


Figure 4. HVAC Systems Data Profile

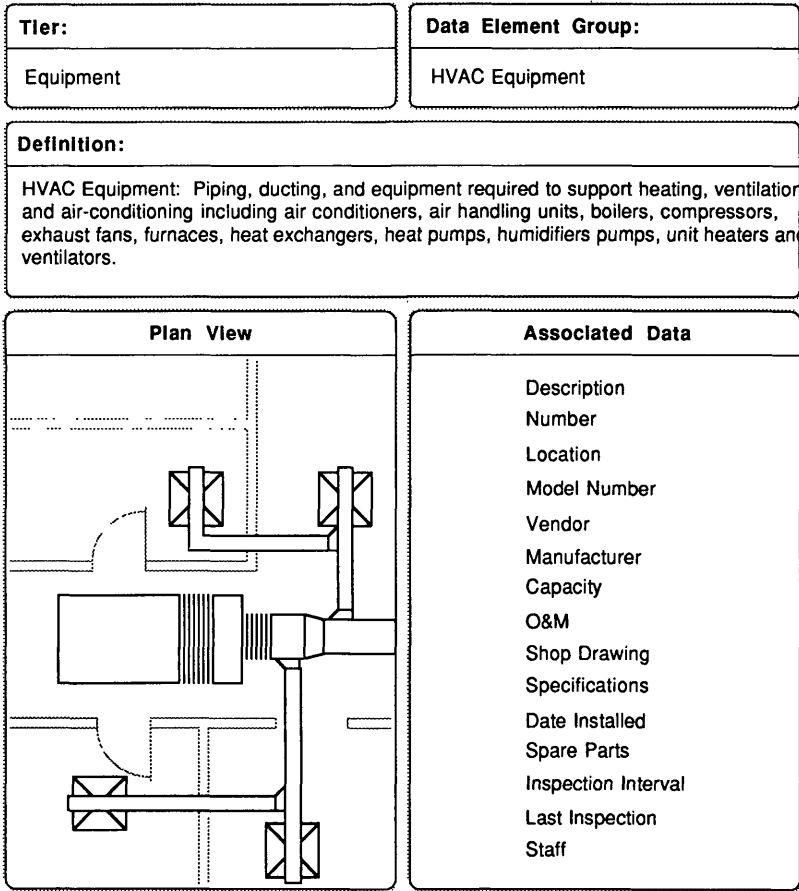


Figure 5. HVAC Equipment Data Profile

INCORPORATING THE LABORDE PROJECTION
INTO AN EXISTING CARTOGRAPHIC SOFTWARE PACKAGE

Piotr H. Laskowski

Intergraph Corporation
One Madison Industrial Park
Huntsville, AL 35807

ABSTRACT

A modular method for incorporating the Laborde projection into an existing cartographic software package, which requires the minimum amount of extra coding, is described. The method exploits the natural modularity in the definition of the Laborde conformal projection and utilizes the existing transformations likely to be found in any standard software library of map projections. The few missing formulas that would be necessary to complete the Laborde mapping equations (but are hard to find in the literature - such as the scale factor equation) are provided. Also, in case the suitable conformal latitude routines are not available, an alternative approach (using Mercator equations) to the transformation from ellipsoid to sphere is proposed.

INTRODUCTION

Cartographic projections enable the representation of the curved surface of the ellipsoidal (or spherical) Earth onto the flat surface of the map. Some projections are used more often than others, mainly based on their usefulness, but also based on the traditionally established standards. Lambert Conic Conformal and Transverse Mercator projections became standards for large scale mapping. These projections are always easy to find in any standard cartographic software package.

Some other projections are almost extinct from today's cartographic practice, often because they were not useful, but sometimes because they never gained enough attention. The Laborde conformal projection, used for Madagascar Grid, is a good example of a projection with the single implementation: for Madagascar only. This projection is not likely to be found in a standard cartographic software package, partly because the computer ready projection equations are not likely to be found in any modern cartographic textbook! Yet the projection has all the desired geometric properties of the Oblique Mercator projection, without being so undesirably sensitive to the small changes in the defining azimuth of the central line, as is the well known Hotine's version of the Oblique Mercator projection (compare Hotine 1947). In many respects, the Laborde approach produces equations that are more numerically

stable than the usual Oblique Mercator's equations.

The cartographer who wants to include the Laborde projection into his cartographic software package has to cope with the old-fashioned, often tabular descriptions (mainly in French and German) of the Gauss-Schreiber projection, a projection similar but not identical to the Transverse Mercator projection.

This paper describes the alternative approach, in which the Laborde projection equations are constructed from the separate functional modules, likely available in any standard cartographic software library. This approach utilizes the intrinsic modularity in the original definition of the projection, and minimizes the amount of new computer code required.

DEFINITION OF LABORDE PROJECTION

The Laborde conformal projection was formulated by Commander J. Laborde (1928) as a triple projection, designed for Madagascar, an island elongated in a direction which is at an angle to the meridian and to the parallel. The properties of the projection are controlled by a set of projection parameters, in a manner similar to the well known Oblique Mercator projection (Hotine 1947):

lat0 - the latitude of origin
(-18°54' for Madagascar)
lon0 - the longitude of origin
(46°26'13.95" for Madagascar)
Az - the azimuth of the axis of strength
(18°54' for Madagascar)
k₀ - the scale reduction factor at origin
(0.9995 for Madagascar)
FE - False Easting
(400000m for Madagascar)
FN - False Northing
(800000m for Madagascar)

Despite the fact that both, Laborde's and Hotine's projections have a similar set of defining parameters, conceptually they have been constructed in a different manner. The Laborde version uses the conformal sphere as an intermediate surface, whereas the Hotine's version uses a special geometric form called aposphere as an intermediate surface.

Computationally, the obliquity of the Laborde projection is controlled by the azimuth at the projection origin, normally at the center of the map, whereas the Hotine's equations indirectly use the azimuth at the equator of aposphere, positioned usually thousands of miles away from the mapped area. As a result, the Laborde equations are numerically more stable with respect to the small variation in the defining azimuth than the Hotine's equations are. Also, Laborde's equations are well defined for azimuth angles close or equal to 0° or 90°, whereas Hotine's equations are not.

As a triple projection, the Laborde equations can be decomposed into three separate conformal projections:

1. ellipsoid to sphere, according to Gauss representation of the second kind (Gauss 1844)
2. sphere to plane, using the spherical Transverse Mercator equations
3. plane to plane, according to the Laborde complex third degree polynomial (Laborde 1928).

The first step defines the intermediate surface of the conformal sphere. The radius of the sphere is the Gaussian Mean Radius of Curvature calculated at the projection origin. The latitude of the projection origin is maintained true to scale (standard parallel), and the scale differs very little from unity in a wide zone surrounding the standard parallel (by design the scale error is maintained close to zero as the quantity of the third order with respect to the angular distance from the standard parallel).

Although the second step alone, the Transverse Mercator projection, does not require any explanation, it is an interesting fact that the first step and second together produce the Gauss-Schreiber projection of the ellipsoid to the plane, which differs slightly from the ellipsoidal Transverse Mercator projection, in that the central meridian is not quite true to scale.

The third step is the conformal transformation from the intermediate plane of the Gauss-Schreiber projection to the final plane of the Laborde projection, designed to reduce the scale error along the chosen oblique axis at the expense of losing an "almost true to scale" meridian generated by the Gauss_Schreiber projection. This is achieved through the (complex) polynomial transformation of the plane, rather than through an ordinary planar rotation.

IMPLEMENTATION STEPS

In this section the software implementation steps for the Laborde projection will be outlined. Only the equations not likely to be found in a standard cartographic software library will be given. These few equations would have to be coded in the form of subroutines and added to an existing software library. The final code for the Laborde projection should then be composed of successive calls to the existing routines, precisely in the order implied by the original definition of the projection.

Step 1. Ellipsoid to sphere

The implementation of this step depends on the availability of the appropriate conformal latitude and longitude subroutine. There are (infinitely) many ways to conformally project ellipsoid to sphere. The Laborde projection specifically requires the application of the Gauss equations of the second kind (Gauss 1844). However,

the most commonly known conformal latitude equations are those used in Adams (1921, p. 18,84), discovered by Lagrange in 1779. The Lagrange representation differs from the Gauss representation of the second kind in that it produces bigger scale errors as the angular distance from the standard parallel increases. Therefore, the Lagrange representation cannot be used for the Laborde projection.

Concluding this step, if the conformal latitude and longitude subroutine which uses precisely the Gauss representation of the second kind is available - it should be used to transform the ellipsoid to the conformal sphere.

Otherwise, the computations in this step may be accomplished, in three separate stages, by the following procedure (which follows directly from the Gauss original definition, and algorithmically utilizes the ordinary Mercator projection equations):

1. The conformal transformation of latitude and longitude (lat,lon) on the ellipsoid to the isometric plane (x,y) may be accomplished by using the forward equations of the ellipsoidal Mercator projection. The parameters to the Mercator subroutine should specify the eccentricity e, the unit equatorial radius a=1, the (Mercator) origin (lat=0,lon=lon0), and the equator true to scale.

In the formulas below, the latitude and longitude coordinates on the ellipsoid are denoted by (lat,lon), the respective latitude and longitude coordinates on the conformal sphere are (LAT,LON), the ultimate origin point of the Laborde projection on the ellipsoid is at (lat0,lon0), and the respective origin on the conformal sphere is (LAT0,LON0).

2. The conformal transformation of the (ellipsoidal) isometric plane (x,y) to the (spherical) isometric plane (X,Y) is accomplished by the Gauss linear equations (Gauss 1844):

$$\begin{aligned} X &= c * x \\ Y &= c * (y + dy) \end{aligned} \quad (1)$$

where the scale and shift parameters should be precalculated as the projection constants:

$$c = [1 + (e^2 \cos^4(lat0)) / (1-e^2)]^{1/2} \quad (2)$$

$$dy = Y0 / c - y0 \quad , \quad (3)$$

where the isometric latitudes y0 and Y0 in equation (3) may again be evaluated using Mercator projection equations.

The ellipsoidal isometric latitude y0 is computed as the Northing value obtained by applying the forward equations of the ellipsoidal Mercator projection to the Laborde

origin (lat0,lon0). The parameters to the Mercator subroutine should specify the eccentricity e, the unit equatorial radius a=1, the (Mercator) origin (lat=0,lon=lon0), and the equator true to scale.

The spherical isometric latitude Y0, needed in equation (3), is computed as the Northing value obtained by applying the forward equations of the spherical Mercator projection to the origin point on sphere (LATO,LONO), where, from Gauss conditions, LAT0 should be computed as

$$LATO = \arcsin(\sin(\text{lat0})/c) , \quad (4)$$

and LONO = 0. The parameters to the Mercator subroutine should specify the eccentricity e=0 (for sphere), the unit radius a=1, the (Mercator) origin (LAT=0,LON=0), and the equator true to scale.

3. The conformal transformation of the (spherical) isometric plane (X,Y) to the resultant latitude and longitude (LAT,LON) on the conformal sphere may be accomplished by using the inverse equations of the spherical Mercator projection. The parameters to the Mercator subroutine should specify the eccentricity e=0 (for sphere), the unit radius a=1, the (Mercator) origin (LAT=0,LON=0), and the equator true to scale.

After the above steps, the resultant (LAT,LON) coordinates refer to the conformal sphere, precisely as implied by the Gauss representation of the second kind (Gauss 1844).

Step 2. Sphere to Gauss-Schreiber plane

For this step the spherical Transverse Mercator equations are appropriate. The parameters to the Transverse Mercator subroutine should specify the radius R which is equal to the Gauss mean radius R0 associated with the conformal sphere, and evaluated at the latitude of origin, lat0:

$$R_0 = a (1 - e^2)^{1/2} / (1 - e^2 \sin^2(\text{lat0})) . \quad (5)$$

Other parameters should specify the eccentricity e=0 (for spherical equations), the (Transverse Mercator) origin LAT=LATO (given by equation (4)), LON=0, and the scale reduction factor at the origin k0=1.

The resultant coordinates on this intermediate plane are precisely the Gauss-Schreiber coordinates of a (double) projection of the ellipsoid on the plane, similar (but not identical) to the ellipsoidal Transverse Mercator projection.

Step 3. Gauss-Schreiber plane to Laborde plane

This step should be programmed in the form of a subroutine implementing Laborde's conformal polynomial equations. These equations will be given here in the

order of calculations.

Given the azimuth Az of the axis of strength (equivalent to the central line in the Oblique Mercator projection), evaluate the projection constants A and B :

$$A = (1 - \cos(2 Az)) / (12 R_0^2) \quad (6)$$

$$B = \sin(2 Az) / (12 R_0^2). \quad (7)$$

Then, for any given Gauss-Schreiber coordinates (x,y) , the mapping equations, which produce the Laborde coordinates (X,Y) , are

$$X = x + A f1 + B f2 \quad (8)$$

$$Y = y - B f1 + A f2 \quad (9)$$

where

$$f1 = -x^3 + 3xy^2 \quad (10)$$

$$f2 = -3x^2y + y^3. \quad (11)$$

Of course, as in any mapping equations, the final X,Y coordinates may be (uniformly) scaled down by the central scale reduction factor k_0 ($k_0=0.9995$ for Madagascar), and the appropriate False Easting, False Northing may be added for the positive coordinates range.

Improving numerical stability

The large numbers that could be possibly generated in equations (10) and (11) may be easily avoided by the following modifications:

a) in Step 2, the call to the spherical Transverse Mercator equations should specify the radius parameter R equal to 1 instead to R_0 of equation (5),

b) in Step 3, the Laborde constants A and B (equations (6)(7)) should be evaluated using $R_0=1$, and the resultant Laborde coordinates of equations (8)(9) should be post-multiplied by the actual R_0 , as properly determined in (5).

NOTE ON INVERSE EQUATIONS AND SCALE FACTOR COMPUTATION

The inverse mapping equations for the Laborde projection should be implemented by using the respective inverse equations for steps 3, 2, and 1 of the forward procedure. Again the assumption is that the inverse equations of the (spherical) Transverse Mercator projection and the Mercator inverse projection equations are available, and should be used in steps 2 and 1 (whenever applicable). The remaining steps require some additional explanation.

Beginning the inverse process with Step 3, the inverse of the Laborde conformal polynomial equations (8)(9) is accomplished by numerically solving for the unknown Gauss-Schreiber coordinates (x,y) , using the given

Laborde coordinates (X,Y) as constants, from the system of nonlinear equations (8)(9), by using the method of simple iteration, also known as the method of fixed-point iteration (Burden, et al 1981). The initial approximation $(x_k, y_k)_{k=0} = (X, Y)$ is appropriate, where (X,Y) denotes the initial Laborde's Easting, Northing coordinates, from which the False Easting and False Northing, the scale factor k_0 , and the Gaussian radius R_0 (equation (5)) were removed. In the case of Madagascar Grid (Laborde Projection Tables 1944), only two iterations are necessary to achieve the required accuracies. However, in the context of this paper, in the general application of the Laborde projection, it is better to allow for as many iterations as necessary for the complete numerical convergence.

In the final step of the inverse Laborde equation (conformal sphere to ellipsoid, the inverse of Step 1), if the Gaussian conformal latitude equations are not available, the ordinary Mercator equations are used again in a precisely inverse order to that described in the forward equations. In this case, the inverse form of the linear equations (1) must be used.

The equations for the scale factor k as a function of lat, lon on the ellipsoid are derived from the fact that a sequence of conformal transformations, performed in succession, produces a conformal transformation with the resultant scale factor equal to the product of the individual scale factors. Again the scale factor equations of the Transverse Mercator projection and the regular Mercator projection (if applicable) are obtainable from any standard software package. The scale factor associated with equation (1) is of course the Gauss constant ratio c given by equation (2). The derivation of the scale factor associated with the Laborde conformal polynomial (8)(9) is only a little more complicated. Using the complex numbers notation, mapping (8)(9) may be written as

$$Z = z + (B+Ai)z^3 \quad (12)$$

where

$$Z = Y + Xi, \quad z = y + xi, \quad i^2 = -1 \quad (13)$$

From the theory of conformal mapping (analytic functions) we have a direct expression for the scale factor at the arbitrary Gauss-Schreiber coordinates (x,y):

$$k(x,y) = [dZ/dz]_{x,y} = 1 + 3(B+Ai)(y+xi)^2 \quad (14)$$

This equation may be programmed using complex arithmetic or treating separately the real and imaginary parts.

Remark: in programming of the scale factor sequence, it is important to always transform the point of evaluation, given at a start as an arbitrary point (lat,lon) on the ellipsoid, to the intermediate surface appropriate for

the transformation component being evaluated.

REFERENCES

Adams, O.S. 1921, Latitude developments connected with geodesy and cartography with tables, including a table for Lambert Equal-Area Meridional projection: U.S. Coast and Geodetic Survey Spec. Pub. No. 130

Burden, R.L., J.D. Faires, A.C. Reynolds 1981, Numerical Analysis, second edition, Prindle, Weber & Schmidt, Boston, pp. 26-32.

Gauss, C.F. 1844, Untersuchungen über einige Gegenstände der höheren Geodäsie, Göttingen Abhandl. 2

Hotine, Brig. M. 1946-47, The orthomorphic projection of the spheroid: Empire Survey Review, v.8, p.300-311; v.9, p.25-35, 52-70, 112-123, 157-166.

Laborde, Chef d'escadron 1928, La nouvelle projection du service géographique de Madagascar: Cahiers du service géographique de Madagascar, Tananarive, No.1

LABORDE PROJECTION TABLES Madagascar 1944, War Department Corps of Engineers, U.S. Lake Survey, New York Office, Military Grid Unit

IBM PC ANIMATION – CRUDE BUT EFFECTIVE

William T. Verts
COINS Department
University of Massachusetts
Amherst, MA 01003

ABSTRACT

Owners of IBM PC's (or equivalent) equipped with the primitive Color Graphics Adapter (CGA) have trouble in creating convincing animated effects. The CGA lacks hardware that allows double buffering. While double buffering is possible on the more advanced adapters, most graphics packages restricted to the CGA are forced to redraw each image directly on the display screen. This is extremely distracting when an animated effect is desired. Experiments with the motion of a vertex through a Delaunay Triangulation show that it is extremely difficult to determine which point is moving when the mesh must be redrawn on the screen after each change in position. This paper presents a mechanism for achieving relatively smooth animation on systems equipped only with the CGA. The technique simulates double buffering by treating an off-screen area of memory as the display area for graphics commands. To then "instantly" update the screen the entire off-screen memory area is copied into the area of memory reserved by the display adapter. Tests using a slow PC show that a sixteen kilobyte image frame can be copied to the screen memory in under one twelfth of a second, sufficient for the illusion of smooth motion.

INTRODUCTION TO THE CGA

Why Use the CGA?

The Color Graphics Adapter was the first and most primitive graphics card produced for the IBM Personal Computer (PC). Although more advanced adapters have been produced and have become popular in later years, many machines are still equipped only with the CGA. Similarly, there is a large software base requiring that an adapter have hardware compatibility with the CGA. Upgrading to a newer adapter (and its corresponding monitor) may be an expense people are not willing to pay. Owners of the IBM-PC Portable (the "luggable", not the lap-top) may also have difficulty in finding a replacement adapter that provides the composite video signal required by the internal monitor.

What Can the CGA Do?

The CGA supports two major text modes: 80 columns by 25 lines and 40 columns by 25 lines. Each character occupies one byte, and all characters are printable. Character definitions conform to standard ASCII for characters in the range 32..126 (the printable ASCII characters), with IBM-PC specific characters for the remainder of the 256 patterns. Associated with each character is an *attribute byte* that describes how the character is to be shown on the screen: characters may be one of 16 colors, on one of 8 background colors, and may blink.

The CGA also supports two graphics modes made up of a rectangular grid of *pixels* (picture elements, or spots of color on the screen), where each pixel may be one of a limited set of colors. The two modes are 320x200 pixels with four colors available per pixel, and 640x200 pixels with two colors per pixel. In addition, there is a little known "unofficial" mode that allows 160x100 pixels with sixteen colors per pixel, but this mode is accessible only by directly programming the video driver chip in the CGA. Note that 320x200 mode, for example, means that visible on the screen are 200 horizontal *raster lines* with 320 pixels per line.

Along with the special plug for the color monitor, the CGA card has an coaxial cable plug which provides a *composite video* signal of the image on the screen. Users could RF-modulate this video signal and use a standard NTSC television rather than being forced to purchase a special purpose monitor (the 25x40 text mode is present to compensate for the limited video bandwidth of most TV's). It is very easy to record the images produced by the CGA on a standard Video Cassette Recorder, as long as the VCR is capable of accepting composite video without the RF component. Composite video is not produced by the higher definition adapters such as the Enhanced Graphics Adapter (EGA).

CGA DISPLAY MODES

The *video memory* for the CGA is mapped onto the address space of the PC as a 16K block starting at absolute address \$B800:\$0000 (16K is shorthand for 16 kilobytes, where a kilobyte is 1024, or 2^{10} bytes). Text and graphics are displayed by storing bytes into that 16K block, which the CGA continuously scans to form the video signals.

Text Modes

A CGA text screen occupies 4K bytes in 80 column mode and 2K bytes in 40 column mode. Half of those bytes contain the characters displayed and half contain the attribute information for each character. The 16K memory reserved for the CGA is partitioned into multiple pages (four 80 column and eight 40 column pages), any of which can be the *active page* (the page being written to) and any can be the *visible page* (shown on the screen). This technique provides a simple mechanism for "instantaneously" changing the screen by writing text into an invisible page, then instructing the CGA to make that page visible (as with the SCREEN instruction in Microsoft's Basic interpreter (IBM; 1982)). Short sequences of text-based animation (four or eight frames depending on the text mode) are created by writing images into each page and then in a loop making each page visible for a short time.

Graphics Modes

In graphics mode the entire 16K block of video memory is used by a single screen, which is always visible. The CGA lacks the memory and hardware needed to support more than one graphics page. There is no native hardware support for animation as there is in the EGA and the more advanced graphics adapters.

For the two native graphics modes the 16K video memory is partitioned into two 8K fields. The first 8K field contains the even numbered raster lines [0, 2, ..., 198] and the second 8K field contains the odd numbered raster lines [1, 3, ..., 199]. Each raster (one line of pixels) occupies 80 bytes, and the raster lines are consecutive within a field (there are no free bytes between rasters). There are 192 unused bytes at the end of each field.

In 640x200 graphics mode each byte represents eight pixels, one bit per pixel. Each pixel has two values: 0 corresponds to black (always the background color), and 1 corresponds to a single *foreground* color selected by the user. This foreground color is one of the sixteen native CGA colors.

In 320x200 graphics mode each byte represents four pixels, two bits per pixel. Each pixel has four values, where 0 corresponds to a user selected *background* color (again, one of the sixteen native CGA colors). Colors 1 through 3 are fixed depending on which of four *palettes* (numbered 0 through 3) has been selected. For example, in palette 0 the four colors are background, green, red, and brown. Although four colors can appear on the screen simultaneously, the user can not arbitrarily select which four are to be displayed.

Selecting the graphics mode and attributes of the screen (palette, background color, and foreground color, as required) are accomplished through calls to the operating system

and by directly setting the CGA hardware registers to the appropriate values (Hogan; 1988) (Crayne, Girard; 1985). Most high level languages for the PC such as Turbo Pascal (Borland International; 1987, 1988) supply procedures to simplify controlling the screen.

Snow

The CGA comes into conflict with the processor of the PC when they simultaneously access the same byte in video memory. This conflict shows up on the screen as flashes of brightly colored "snow" in text modes and in the 160x100 graphics mode (which the PC still "thinks" is a text mode). Snow can be eliminated under software control by waiting to store bytes into video memory until the horizontal or vertical video retrace interval. This technique significantly slows screen operations, but because it is a software technique it can be switched on and off as desired. Some new implementations of the CGA do not suffer from snow.

CGA ANIMATION

Borland International has included with recent versions of Turbo Pascal, Turbo C, Turbo BASIC and Turbo Prolog routines that provide device independent graphics support for a large number of graphics adapters (Borland International; 1987, 1988). Software support for animation is present if multiple screens are available in the hardware. There is of course no such support for the CGA.

Although DOS (the operating system) contains functions for manipulating pixels in the video memory, a graphics package can be written to modify "virtual pixels" in any 16K block of memory according to the storage rules listed earlier. A new image prepared in an off-screen *frame* becomes visible when it is copied to the video memory of the CGA.

Time and Space

"Active" animation frames reside in main memory data-structures, so copying one frame to another is strictly a memory-to-memory transfer. Tests on a 4.77Mhz PC/XT show that one frame can be copied to another in about one-twelfth of a second. Machines with a higher clock rate can copy frames even faster. In addition, the PC/AT can make most advantage of its 16-bit bus to increase transfer speed if two conditions are met: frames must be word aligned in memory (the base address is an even number), and the frame-copy code must transfer 16-bit words instead of 8-bit bytes.

Each frame occupies 16K bytes of memory. Main memory is restricted on the PC to 640K bytes, of which portions must be reserved for DOS, device drivers, main memory RAM-disks, any TSR programs (Terminate and Stay Resident, "pop-up" programs), and the animation program itself. Forty frames would completely fill the 640K bytes of main memory; the practical limit is much lower. A 640K system without TSR's or a RAM-disk can generally support a program requiring 22 frames, which is exactly what a 360K byte, 5 1/4 inch diskette can hold.

An image can be loaded into memory from a 360K floppy disk in about one second, and from a hard disk in about one-fifth of a second, depending on the quality of the disk drives.

Systems equipped with EMS (Expanded Memory) can allocate a very large chunk of EMS memory as a RAM-disk and keep the bulk of unused images stored there "off line". A copy from RAM-disk is orders of magnitude faster than pulling the images off of magnetic media.

PASCAL CODE EXAMPLES

Basic Techniques

In (Adams; 1988) a few off-screen CGA frames are located at fixed places in high memory. This technique limits the number of frames available to those defined at compile-time and permits conflicts to occur between image frames and code that may accidentally extend into the reserved areas. By dynamically allocating memory for frames from the Pascal heap instead, the number of off-screen frames available can be determined at run-time. All code examples that follow are written in Borland's Turbo Pascal and are compatible with version 3.0 or later.

Two type definitions are critical to the construction of animation frames: an array type of the correct size (16K) and a pointer type that points to objects of that array type. The Pascal code fragment below shows those type definitions and two variables of the pointer type.

```
Type      Screen      = Array [0..16383] Of Byte ;
          Screen_Pointer = ^Screen ;

Var       A, B          : Screen_Pointer ;
```

Screens are allocated for variables A and B of type Screen Pointer from the heap with standard Pascal procedure New. The expression New(A) allocates a 16K block from the heap and assigns the address of that block to A (in Turbo Pascal a function can be installed to return Nil if an allocation attempt fails, rather than causing the program to abort). Alternatively, the expression A := Ptr(\$B800:\$0000) directly assigns to variable A the address of the CGA video memory. Copying screen B into screen A is accomplished by the simple assignment statement A^ := B^.

The Frame Handler

Although a single off-screen memory block is sufficient to perform all feats of graphics animation, many animation tasks are simplified by having two or more off-screen frames. The definition below is of an array of pointers to animation frames, and an index variable that indicates which of those frames is *active* (being issued graphics commands).

```
Var       Active_Table   : Array [0..40] Of
                                Screen_Pointer ;
          Current_Active  : -1..40 ;
```

The Active_Table array can be made as large as necessary, although the limit of 40 insures that there are enough entries to completely fill the 640K system memory with frames. All entries in Active_Table are initialized to Nil. When a off-screen frame is *opened*, memory is allocated for it from the heap and its address is assigned to the proper entry in Active_Table. When a frame is *closed*, its memory is returned to the heap with Pascal procedure Dispose, and the corresponding entry in Active_Table is set back to Nil.

By definition, Active_Table[0] always refers to the visible screen; its value is either \$B800:\$0000 (if opened and in graphics mode) or Nil (if closed and in text mode).

Current_Active is the index into Active_Table of the active frame. The active frame must be open (i.e., when Current_Active is greater than or equal to zero Active_Table[Current_Active] will not be Nil). Current_Active is initialized to -1, and otherwise equals -1 when no screen is active (several may be opened, but none will receive graphics).

Copying Frames

Any open screen may be copied into the currently active screen by the expression: `Active_Table[Current_Active]^ := Active_Table[Target]^`. Animation is accomplished by copying a screen into the visible screen (`Active_Table[0]`).

Clearing Frames

The active frame can be cleared with the expression: `FillChar(Active_Table[Current_Active]^, SizeOf(Screen), #0)`. This expression uses Turbo Pascal routine `FillChar` to flood the 16K block with zeroes (`#0` is the character with ordinal value zero). Frames may be initialized to other values by changing the flood character.

Loading and Storing Frames

It is very useful to save images to and load images from disk. A code fragment for storing the image in the active frame is shown below; loading images from disk requires similar code. `File_Name` must contain a valid DOS file name.

```

Type      Frame_File      = File Of Screen ;

Var       File_Name       : String ;
          Outfile         : Frame_File ;

Assign   (Outfile, File_Name) ;
Rewrite  (Outfile) ;
Write    (Outfile, Active_Table[Current_Active]^) ;
Close    (Outfile) ;
```

Painting Pixels

The most important action that can be performed on a graphics screen is to set a particular pixel to a particular color. For the CGA, with each raster line occupying a contiguous group of bytes, this task is broken into three phases: determining the location of the correct raster line, locating the correct byte within that raster, and setting the correct pixel within that byte.

Assume that variables `X` and `Y` contain the coordinates of a visible point in the active frame, and that the pixel will be set to the value in `Color`. A code fragment to set the pixel to the desired color is given below. Variables `X`, `Y`, `Offset` And `Index` are of type `Integer`, variables `Color`, `Mask`, `Shift`, `Pixel` and `Image` are of type `Byte`.

The code is identical for both 320x200 mode and 640x200 mode except for the four lines marked with the `(**)` comment. Those lines each contain pairs of adjacent numbers where the rightmost of each pair is commented out with curly braces. The code is set up for 320x200 mode. To change to 640x200 mode, delete the curly braces and place them around the leftmost number of each pair.


```

(* Compute the offset of the correct raster *)
Offset := (Y Div 2) * 80 ;
If Odd(Y) Then Offset := Offset + 8192 ;
(* Compute the offset of the correct byte *)
Offset := Offset + (X Div 4 { 8 }) ;      (!!)
(* Determine pixel position within byte *)
Index := (X Mod 4 { 8 }) ;              (!!)
(* Invert Index and scale by bits per pixel *)
Shift := (3 { 7 } - Index) * 2 { 1 } ;  (!!)
(* Determine mask to modify correct pixel *)
Mask := ($03 { $01 } SHL Shift) ;      (!!)
(* Build colored pixel to go in new place *)
Pixel := (Color SHL Shift) AND Mask ;
(* Get image byte from active screen *)
Image := Active_Table[Current_Active]^[Offset] ;
(* Modify the pixel *)
Image := Pixel OR (Image AND NOT Mask) ;
(* Replace image byte back into active screen *)
Active_Table[Current_Active]^[Offset] := Image ;

```

This code fragment is very inefficient. Execution speed can be improved by replacing the computations of `Offset` and `Mask` with table look-ups of precomputed values. Expressions containing `Div` and `Mod` by powers of two can be replaced with shifts. Many of the assignment statements can also be condensed into a few long statements (variable `Image` is not even necessary).

Drawing Lines

The Bresenham algorithm is the classic algorithm for painting lines between any two points. The algorithm works by stepping one pixel at a time from one endpoint of the line to the other, painting a spot of color at each pixel. This algorithm can be easily written in assembly language because it uses integers rather than real numbers to determine the deviation of the painted line from the true slope. A FORTRAN implementation of this algorithm appears in (Bowyer, Woodwark; 1983).

Using the Bresenham algorithm is inefficient when drawing horizontal lines on the CGA. A horizontal line routine can take advantage of the knowledge that each byte in an image frame contains several pixels from the same raster line, and that a horizontal line is contained in a contiguous group of bytes. The first and last bytes of the group (containing the left and right endpoints of the line) must be masked off so only pixels that are part of the line are changed. Bytes in-between the first and last bytes are completely part of the line, and those bytes all receive the same value: a byte where all pixels have the desired color.

USAGE AND EXAMPLES

This section presents several applications of the frame animation technique. All of the examples listed here have been implemented on a standard 4.77Mhz IBM PC/XT equipped with a CGA and an 8087 numeric coprocessor. Each example demonstrates one or more places where standard CGA graphics is inferior to the frame animation technique.

Real-Time #1: Removing Plotting Distractions

An experiment was conducted to observe what happens to a Delaunay Triangulation as one of the vertices moves through the plane. When the mesh is repainted directly on the screen between iterations, it is extremely difficult to visually isolate the moving point from those that are not moving. Repainting gives an illusion of motion to lines that have not changed positions between iterations. When the graphics are painted into an off-screen frame, then "instantly" copied to the visible screen, the motion of the

point becomes quite apparent. On a 4.77Mhz machine, painting the graphics is slower than generating the Delaunay Triangulation for up to 20 points, even using an $O(N^4)$ mesh generation algorithm.

Real-Time #2: Illusion of Motion

Three dimensional, near real-time rendering of simple molecular models can be performed on a PC. Atoms are represented by spheres, which have the characteristic that after scaling, rotation, and perspective transforms are applied a sphere still looks like a filled circle. The molecular model is rotated, then atoms in the model are sorted according to distance from the viewer. Filled circles are painted into the off-screen frame from the atom that is furthest away to the atom that is closest. The radii of the circles depend on the relative sizes of the corresponding atoms and on how far they are from the viewpoint. Atoms that are small or are far away tend to be obscured by large or nearby atoms.

When the image is completely rendered it is copied to the screen. The time between updates is dependent on the complexity of the molecular model and the sizes of the circles that must be drawn, but small molecules can be processed in just a few seconds. The short pause between complete frames is far less distracting than the "popcorn effect" of having the circles plotted directly on the visible screen for each iteration.

Real-Time #3: Time Constraints

A graphic simulation of an analog clock (one with hands) requires that the screen be updated once every second. For this task there need to be two off-screen frames. One frame contains the background image of the clock face, which does not change, and the other is a work screen needed to generate each new clock image. The background image is copied to the work frame, the hands are added in their new orientation, and the result is then copied to the visible screen. Despite the fact that the process requires two 16K frame copies, there is ample time to generate each new image within the one-second time limit.

Menus and Rubber-Banding

A mouse driven graphics paint program can take advantage of the animation facilities in several ways. The work image and the menu image are kept in separate frames so that the menu does not interfere with the drawing being generated. When the menu image is needed the work image is copied off to a temporary frame and the menu image is copied to the visible screen. When the menu action is complete, the work image is copied back to the screen.

Keeping an "Undo" screen is trivial. By saving the work image before each new operation is performed, the undo frame always contains the last valid image and is always available in case a mistake is made.

The animation facility is fast enough to support *rubber-banding*. When drawing lines, boxes or other objects it is useful to display the new object moving or changing in size according to the position of the mouse until the size and position of the object are satisfactory. This can be accomplished by saving the work image, then as the mouse moves the saved image is copied to a temporary area where the figure is added in its new orientation, and the result is then copied to the visible screen. When the correct position has been determined, the figure is added permanently to the image.

Preprocessed Frame Movies

When several images of a scene have been rendered where each is slightly different from its predecessor, the images can be copied into the visible screen area fast enough to create the illusion of smooth motion. A sequence of 20 images of the Earth, where in

each successive image the Earth has been spun by 18 degrees, will create the illusion of a smoothly spinning planet.

The motion will be a lot coarser if there are more animation images than will fit into main memory. For example, a 64 image sequence occupies one megabyte of memory and each frame must be loaded successively off of the hard disk. Loading from the hard disk is a lot slower than moving an image around in main memory.

Mandatory Off-Screen Graphics

If the PC has no CGA compatible graphics adapter, graphics images can still be produced in off-screen frames and saved for display on another machine.

CGA EXTENSIONS

A graphics package is minimally complete if it contains procedures for setting any pixel to any allowable color and for examining any pixel to see what color it contains.

Clearing the screen, drawing lines between any two points (clipping those lines to the screen if necessary) and copying chunks of an image from one place to another can be completely described in terms of setting and examining individual pixels, but these tasks can often be optimized by carefully considering how image memory is organized.

An animation graphics package must also be able to allocate and de-allocate frames, move images from frame to frame, load images from disk, store images onto disk, and make any of the frames the active frame for graphics commands.

In addition to the techniques presented here, there are several ways to extend the capabilities and usefulness of the graphics package.

Graphics Modes

Other than the three graphics modes that can be displayed on the CGA monitor (640x200 two color, 320x200 four color, and 160x100 sixteen color), graphics modes can be supported that cannot be shown on the CGA monitor. It is a simple intuitive step to realize that off-screen graphics techniques can be used for formats the CGA does not naturally support, so long as no attempt is made to directly display images in those formats. An "image processing" mode, for example, is 128x128 pixels with 256 colors per pixel (each pixel occupies one full byte). This format contains the largest square that will fit into a 16K byte CGA frame ($16384 = 128^2$). All graphics primitives can be executed on frames in this mode, except that frames may not be copied to the video memory. The other modes become very useful for previewing, as long as one is willing to accept restrictions on the number of colors available or the size of the visible area.

Circles and Ellipses

A routine that draws a circle is useful. An ellipse routine is also useful to have, but is a somewhat more complex algorithm than the circle. In addition to the Bresenham line routine, (Bowyer, Woodwark; 1983) also contains a simple circle drawing routine that uses only integers. Integer routines can be efficiently coded in machine language for high speed. An integer ellipse routine appears in (Van Aken; 1984). These routines paint only the outline of the circle or ellipse, but both can be modified to paint solid figures. The ellipse can be used in place of the circle routine by selecting identical values for the major and minor axes, and it can also be used to paint *visually correct* circles on screens that do not have a 1:1 aspect ratio.

Color Mixtures and Patterns

Startling color mixture effects can be achieved with little extra overhead by painting figures with patterns instead of with solid colors. Pseudo-colors can be created by mixing two or more colors in a checkerboard. This is particularly useful on the CGA because of the limited number of available colors.

It is very simple to define patterns as screen invariant so that the color of every pixel can be determined uniquely from its pattern, regardless of what objects are to be painted on the screen. Drawing lines or figures then “uncovers” the pattern in the shape of the drawn object. This technique results in strange visual effects: animation on a screen invariant background pattern more resembles object shaped windows moving over the pattern than patterned objects in motion. It is more complex to write an efficient pattern handler that is object relative than one that is screen relative.

CONCLUSIONS

The CGA (what an acquaintance refers to as the Brain Damaged Adapter), is indeed a primitive graphics standard. Owners of IBM PC's equipped with more advanced adapters, and those who have machines specifically designed for animation, may be amused at the thought of doing animation on the CGA. There are much better ways of performing animation than with the CGA, but when it is the only choice it can be teased into producing spectacular effects.

BIBLIOGRAPHY

Adams, L., 1988. High-Performance Graphics in C — Animation and Simulation, Windcrest Books (a division of TAB Books), Blue Ridge Summit, PA

Borland International, 1987. Turbo Pascal 4.0 Owner's Handbook, Borland International, Scotts Valley, CA

Borland International, 1988. Turbo Pascal 5.0 User's Guide, Borland International, Scotts Valley, CA

Borland International, 1988. Turbo Pascal 5.0 Reference Manual, Borland International, Scotts Valley, CA

Bowyer, A., Woodwark, J., 1983. A Programmer's Geometry, Butterworths, London

Crayne, C. A., Girard, D., 1985. The Serious Assembler, Baen Enterprises, New York, NY

Hogan, T., 1988. The Programmer's PC Sourcebook, Microsoft Press, Redmond, WA

IBM, 1982. BASIC, Reference Manual

Van Aken, J. R., 1984. An Efficient Ellipse-Drawing Algorithm: IEEE Computer Graphics and Applications, Volume 4 #9 (September 1984), pp. 24-35

TRADEMARKS

MS-DOS and Microsoft are registered trademarks of Microsoft Corporation.

Turbo Pascal, Turbo C, Turbo Basic, and Turbo Prolog are registered trademarks of Borland International, Inc.

IBM, IBM PC, IBM Personal Computer, IBM Personal Computer XT, IBM Personal Computer AT, IBM Portable Personal Computer, and PC-DOS are registered trademarks of International Business Machines Corporation.

CAD: A VIABLE ALTERNATIVE FOR
LIMITED CARTOGRAPHIC AND GIS APPLICATIONS

Robert C. Anderson
Lloyd D. Carmack, Jr.
Department of Geography and Computer Science
United States Military Academy
West Point, New York 10996

BIOSKETCH

Major Robert Anderson and Captain Lloyd Carmack are Assistant Professors in the Department of Geography and Computer Science at the United States Military Academy, West Point, NY. Both graduated from West Point with BS degrees in 1974 and 1977, respectively. Major Anderson is an Intelligence Officer with MA degrees in International Relations from the University of Southern California and in Geography (Remote Sensing) from the University of Georgia. Captain Carmack is an Air Defense Officer with a MS degree in Geography (Computer-Assisted Cartography) from Rutgers University. Both are involved with research pertaining to military applications of GIS.

ABSTRACT

The most popular GIS often require extensive investments of hardware, software and training, and may offer capabilities not needed by low end users. Consequently, those users with limited mapping requirements often cannot justify purchasing a GIS. PC-based CAD systems have a viable role in quickly and inexpensively performing limited mapping of layered spatial data, in lieu of GIS. This paper examines how low priced CAD packages can be used in the situations where layering of spatial data is more important than conducting manipulations on the data attributes. The user faces several problems when substituting a CAD package for a GIS. Issues investigated include constructing the data layers from various sources, maintaining registration, updating information and plotting output to a specific scale. CAD, with its inability for in-depth data analysis, in no way substitutes for GIS, but it can serve the low end user as a first step towards a GIS. CAD also has a role as an inexpensive educational tool capable of introducing students to the GIS attributes of inputting, layering, updating and outputting spatial data.

INTRODUCTION

Geographic Information Systems (GIS) continue to be a hot topic and the cornerstone of an ever growing industry. The latest issue of any periodical associated with the field of geography is sure to include at least one article devoted to the subject. Estimates of worldwide revenues for the GIS market are expected to reach \$464 million by 1991 (Lang, 1988). The move from theory to application has also resulted in a more focused definition of GIS to 'a decision support system involving the integration of spatially referenced data in a problem solving environment' (Cowen, 1988). What sets GIS apart from other automated mapping systems are the processing

capabilities related to the encoding, storage, analysis and display of spatial data (Berry, 1985).

As can be expected, the power of a full fledged GIS is often related to its price. Industry leaders such as ESRI, Intergraph, IBM, Synercom and ERDAS can provide complex, multi-user systems which might cost as much as \$400,000 (Lang, 1988). While these systems are on the cutting edge of GIS technology, users having limited GIS or mapping requirements often cannot justify purchasing such a system.

Although not a GIS, Computer Aided Design (CAD) systems are recognized as serving a useful role in the GIS world. While most realize that CAD is limited in both terms of analysis methods and the volume of data that can be handled (Burrough, 1986), CAD is well suited to its role as a tool for cartographic applications due to the inherent electronic drafting and graphic overlay capabilities (Cowen, 1988). For the low end user or educator, CAD's capability to quickly layer spatial data, at relatively low cost, gives it a viable role in lieu of a full fledged GIS.

BACKGROUND

The geography program at the United States Military Academy (USMA) includes courses in Remote Sensing, Photogrammetry, Surveying, Cartography and Computer-Assisted Cartography (CAC). New in the Spring of 1989 is a GIS course. The Department currently has several GIS packages used for research, such as PC ARC/INFO, VAX-based INFORMAP III and GRASS on a Sun workstation. These systems are too complicated and expensive to support teaching an undergraduate GIS course.

Our CAC course uses PC and mainframe mapping programs to introduce cadets to automated cartography. One block of the course requires cadets to produce a map using CAD as a drawing tool, instead of a canned mapping program. Once the cadets have completed this portion of the course, we then use CAD to build interest in the upcoming GIS course by introducing cadets to the GIS attributes of inputting, layering, updating and outputting of spatial data. We accomplish this with low cost (less than \$400) CAD programs (DRAFIX and CADKEY) on IBM AT or Zenith Z248 machines. Our peripherals included Summagraphics digitizing tablets and hardcopy devices such as Alps 2000 printers and IBM and HP pen plotters.

The literature has numerous references to the theoretical use of CAD as a mapping tool and increasingly as a surrogate (or at least a limited substitute) for GIS (Burrough, 1986; Moynihan, 1987; Cowen, 1988). However, the literature failed to prepare us for the problems encountered implementing CAD as a limited GIS. This article focuses on three areas: data capture, data processing and information output/display. We will share some of the issues necessary to consider when using a low cost CAD as a tool for certain mapping and limited graphic overlay applications.

DISCUSSION

Data Input

Because CAD packages are primarily electronic drafting tools, they may not be initially suited for the task of transferring spatial

data from maps and air photographs to digital form. In fact, the inexpensive and relatively simple packages that we employ, use the digitizing tablet as a sophisticated pointing device. Although these packages are laden with commands to implement the many drawing capabilities they possess, they both lack built-in digitizing routines. Consequently, we had to supplement our CAD programs with stand alone digitizing software.

Although a wide range of hardware is available for data input, internal factors such as budget, ease of operation and limited training time available for our students, we limited our efforts at data capture to manual digitizing. A public domain digitizing program, DIGITIZE, developed by the Department, accomplishes basic digitizing operations. Additionally, the program converts coordinate files into the appropriate drawing interchange files which both CAD packages can import as drawing files (Loomer, 1987). While more elaborate CAD programs like AUTOCAD include digitizing routines, the higher price and increased difficulty of use may limit their application by the low-end user or educator. DIGITIZE may not accomplish the more elaborate functions of file editing and line smoothing that are found in a full fledged GIS or a more comprehensive CAD, but it does succeed in transforming the digitizing tablet into a data capture device. It is important, from the stand point of efficiency and error reduction, to digitize input once with a single digitizing program and then convert the files to the various formats required by the different CAD programs.

Other data input considerations associated with the use of CAD are related to both the actual digitizing process and to the manner in which CAD will be used to display and overlay the digital files. Two approaches may be used to exploit the graphic layering capability of CAD. The first approach would capture all the input data features onto a single layer, while the second method would digitize various data features onto individual layers. Experience has shown that more complex input is best reduced into layers prior to digitizing. Items can be moved from layer to layer within CAD; however, this can be a difficult and time consuming proposition if there are many features. We think it is more efficient to separate features prior to digitizing, then merge them as required. It is important to include plenty of reference (or control) points as part of each layer's digitized file.

Data Processing

CAD is an effective graphics display tool because it can turn on or off different layers (of various geographical features) for selective editing and display. Even though CAD does not provide the analytical functions of a GIS, the layering capability can provide a graphics display similar in concept to the multiple layers of a GIS. Exploitation of the CAD layering capability depends on the operator's ability to develop the various layers.

A GIS would make use of techniques similar to INFORMAP's 'Facetization' command which transforms arbitrary source documents into a fixed database (Synercom, 1988). This layering is fairly straight forward in CAD if multiple control points are included during the digitizing process. The process is only slightly more complex if the input sources vary in size, scale or coverage. With CAD, registration to a specific scale is established to a base layer (which may only contain the reference or control points) using

editing functions such as 'Scale' and 'Rotation.' Once registration is established, the layer can be turned off or left on and a new layer of information can be added to the drawing.

Data Output

A problem that plagues anyone working with GIS and CAD is that of transferring the image on the monitor to paper without losing either resolution or information. We checked our output for geometric accuracy to determine the distortion one can expect from mapping with CAD. The digitizer sends coordinates of the input data to the CAD program which in turn displays the output to the monitor using screen coordinates. The scaling and rotation factors we applied to the layers we wanted to register and merge, were based on what we saw on the monitor. The final output, on paper or mylar, was set to either a 1:1 ratio or a multiple of the input (digitized) scale. It is not surprising that some error is introduced as the input coordinates are redefined several times before hardcopy output.

We used CAPTURE from the Desktop Digitizing Program (distributed by R-Wel, Inc. of Athens, GA) which calculates a least squares rectification from an affine solution if four or more control points are used. This program provides the residual error for each control point and the RMSExy (root mean square error) vector error for all points in the solution. It also calculates the error in ground and map units and the overall scale (DDP, 1988).

Output accuracy is a function of preciseness of the input control points. A DMA 1:25,000 topographic map served as the base map for the first test. We digitized map information at 1:25,000 scale, added additional layers of input data at the same scale and output the combined map at both 1:25,000 and 1:50,000 scales. This file had four control points: two were UTM grid line intersections and two were major road junctions. Our DIGITIZE software limited us to inputting the coordinates from the keyboard of only two control points (the grid line intersections). The coordinates for the two road junctions were read using a 1:25,000 grid coordinate scale (Table 1).

TABLE 1
Accuracy Assessment Using Grid Line Intersections
and Road Junctions as Control Points

Data Source	Ctl Pts	Output Scale	RMSExy (m)	% Scale Error
Map				
1:25,000 DMA Topo	4	1:25,061	8.297	-
1:50,000 DMA Topo	4	1:49,981	21.787	-
CADKEY				
1:25,000 at 1:25,000	4	1:25,190	3.80	0.515
1:25,000 at 1:50,000	4	1:50,317	4.58	0.672
DRAFIX				
1:25,000 at 1:25,000	4	1:25,079	3.963	0.0718
1:25,000 at 1:50,000	4	1:49,938	4.155	0.0860

The error in the map data is caused mostly by the inaccuracies of the road junction coordinates. The scale and RMSExy values are the average of three iterations of the DDP. Percent Scale Error was

calculated by dividing the map scale by the difference of the map scale and the output CAD scale.

For the second test we used the same 1:25,000 scale base map and a 1:15,000 orienteering map as an additional source of information. The orienteering map was similar to an aerial photograph since it contained detailed map information and lacked a coordinate grid system (Table 2). We merged the map information of the two maps and printed/plotted output at the scales of 1:25,000 and 1:15,000.

TABLE 2
Accuracy Assessment Using Road Junctions
as Control Points

Data Source	Ctl Pts	Output Scale	RMSExy (m)	% Scale Error
MAP				
1:25,000 DMA Topo	5	1:25,163	20.682	-
1:15,000 Orienteer	5	1:15,308	25.686	-
CADKEY				
1:15,000 at 1:15,000	5	1:15,514	24.634	1.346
1:15,000 at 1:25,000	5	1:25,925	26.653	1.025
1:25,000 at 1:25,000	5	1:25,421	19.758	3.028
DRAFIX				
1:15,000 at 1:15,000	5	1:15,413	19.404	0.686
1:15,000 at 1:25,000	5	1:25,212	25.641	0.195
1:25,000 at 1:25,000	5	1:25,475	20.436	1.240

It is not surprising that the RMSExy error vector and Percent Scale Error increased when using five true "ground" control points, instead of grid line intersections. Other factors which affect the accuracy of the data captured include various hardware and operator errors indicative to the digitizing process (Cameron, 1982).

APPLICATIONS

Cartographic Applications

CAD has been used quite extensively as a mapping tool. Some of the primary applications have been in land planning and planimetric mapping, such as residential subdivisions and commercial areas (Cowen, 1988). CAD is well suited for generating smooth curves in cul-de-sacs, for providing labels, titles and legends and for high lighting features by shading and/or patterns. CAD loses no map information due to generalization when the scale is decreased and it maintains data fairly accurately.

By carefully sizing numbers and labels, the cartographer may choose, for example, to "lose" labels containing square footage but keep house numbers visible on a map of an entire housing subdivision. A separate layer, of a smaller area at a larger scale, may contain lot dimensions and other information not needed at smaller scales. In our West Point phone books, we have several pages of maps detailing the family housing areas. Last year one of our cadets developed such a database. He produced an electronic map of the housing areas, any portion of which could be zoomed in on for detailed information, as well as separate layers containing the various housing areas at larger scales (Albert, 1987). These separate layers could easily be updated and output for future phone book editions.

GIS Applications

Inexpensive CAD compares favorably with GIS systems in several areas. These include cost (hardware, software and operator training), layering of data, updating/adding layers, data display (in vector format) and data output. CAD can be used to overlay layers which enables an operator to visually (manually) identify areas that are common, while this is done by automatically in a GIS. However, CAD cannot manipulate or conduct extensive analysis of attributes on a layer or among layers.

As an example, let us choose to overlay soils, vegetation and road data. We task the GIS to display locations where specific soil types intersect with particular vegetation types and then overlay the roads. The resultant map contains only the areas where the polygonal data are common, with the road data included for reference. This can also be accomplished with CAD. We first assign patterns for our polygonal data to ensure adequate visibility of layers, once they are activated, so data is not masked (a problem using only colors). Or we could put each type of soil and each type of vegetation on individual layers and activate the layers with the particular attributes as necessary. Some minimal analysis of layered data is possible in CAD, but it is easy to see this quickly becomes a labor intensive effort for limited results. Newer and more expensive CAD programs link layers to a database. This only allows the user to tag items on layers with their attributes, not perform manipulations on the layers. As layers are moved, merged and scaled, their attributes follow along; attribute values are not changed or modified in the new layers.

CONCLUSION

Low cost CAD is a high powered drawing tool with many cartographic applications, especially in the field of planimetric mapping. CAD cannot substitute for a GIS, but it can serve the low end user whose requirements are more concerned for layering of spatial data, not for the manipulation of data attributes found on the layers. It is also viable as an introductory tool to GIS in an educational environment, exposing students to the concepts of inputting, layering, updating and outputting spatial data. We realize that by opting for a more expensive CAD package, the operator may experience fewer problems in the data input and layering manipulations. In any case, the use of CAD in a GIS role is not feasible for data manipulation, but limited mostly to constructing, updating, displaying and outputting layered spatial data.

NOTE: Commercial products are described to support the discussion. Their mention does not represent an endorsement by the US Military Academy or the Department of the Army.

REFERENCES

- Albert, D.S., 1988, Final Report On Independent Study Into CAD. Unpublished Research Paper, Department of Geography and Computer Science, West Point, New York.
- Berry, J.K., 1985, Computer-Assisted Map Analysis: Fundamental Techniques. Computer Graphics 1985 Conference, #SIS-85-1, National Computer Graphics Association, Fairfax, Virginia, 18pp.

Burrough, P.A., 1986, Principles of Geographical Information Systems for Land Resources Assessment. Oxford University Press, Oxford.

Cameron, E.A., 1984, Manual Digitizing Systems. Basic Readings In Geographic Information Systems, (ed. Marble et al). SPAD Systems, LTD, Williamsville, New York, pp. 3-11 - 3-18.

Center for Remote Sensing and Mapping Science, 1988. Desktop Digitizing Package Reference Manual, Department of Geography, University of Georgia, Athens, Georgia.

Cowen, D.J., 1988, GIS Versus CAD Versus DBMS: What Are the Differences?, Photogrammetric Engineering and Remote Sensing. Vol. 54. pp. 1551-1554.

Lang, L., 1988, The Movers and Shakers of GIS. Professional Surveyor, Vol. 8, pp. 4-9

Loomer, S.A., 1987, DIGITIZE. Digitizing Tablet Support Program. Computer Graphics Laboratory, Department of Geography and Computer Science, West Point, New York

Moynihan, R., 1987, Micro-Based CAD Systems in Surveying and Mapping Education. Technical Papers, 1987 ASPRS-ACSM Annual Convention, Vol. 4, pp. 133-139.

Synercom Technology, Inc., 1988, INFORMAP III Systems Reference Manual, Version 1.2, Houston, Texas.

ESTABLISHING A CORPORATE GIS DATABASE FROM MULTIPLE GIS PROJECT DATA SETS

Timothy R. Johnson
Karen C. Siderelis
Land Resources Information Service
P.O. Box 27687
Raleigh, North Carolina 27611-7687

ABSTRACT

The Land Resources Information Service (LRIS) has operated the geographic information system (GIS) for the State of North Carolina since system implementation in 1977. LRIS was established to provide the State with a geographic data capture and analysis capability to support natural and cultural resource management. LRIS has conducted several hundred projects over the past 12 years utilizing the capabilities of the GIS to build specialized data sets and perform various analyses with those data sets. LRIS has reached a point where its data resources and the associated investment in the capture and maintenance of that data have become very significant and the use of data is at times unwieldy. The need to establish a statewide "corporate database" of geographic information has become an important issue affecting further growth and the ability to respond effectively to new projects. This paper addresses the background and issues related to the decision to establish a corporate database, the methodology used to design the database, the implementation status to date, and the anticipated benefits of completing such an effort.

BACKGROUND

Recognition of the need for an information system to support geographic-oriented decision making in North Carolina arose out of the North Carolina Land Policy Act of 1974 (Tribble and Siderelis, 1988). This legislative mandate led to the establishment of LRIS as an agency within state government and to the purchase of geographic information system hardware and software from Comarc Design Systems in 1977. Over the years, LRIS performed a wide variety of project work on a cost recovery basis with only a minimal level of state appropriations. These projects can generally be described as one-time efforts for diverse geographic areas (e.g., individual counties or application-specific portions of the state) using a wide variety of base maps ranging in scale from 1:24,000 to 1:250,000. The data layers captured as part of these projects also varied considerably. Typical base layers such as transportation, hydrography, and political boundaries were captured as well as more application-specific layers such as primary and secondary nursery areas for fish and historic and archaeological sites.

During the course of growth of LRIS, GIS technology changed tremendously with advances in both the power and utility of hardware and software to support spatial problem solving. LRIS eventually outgrew the capabilities of the Comarc software and purchased ARC/INFO in 1986. This decision prompted the organization to look at its investment in data captured using the Comarc system and the desire to maintain as much of that data as possible within the ARC/INFO environment. It also was an opportunity for LRIS to develop initial concepts for the design of a corporate database.

A corporate database is one that is established as a corporate or organizational resource as well as one that meets several criteria. The criteria used to guide the design of the corporate database are given below.

- The database should be practical to maintain in terms of institutional and digitization cost considerations.
- The database should consist of data that satisfy common needs.
- The database should be structured to maximize its utility.
- The database should not include all digital data holdings (i.e., not every data layer in digital form should be part of the database).

GOALS AND ISSUES

LRIS is a service organization whose mission is to maintain a digital database of geographic information for the State of North Carolina and to provide GIS services to a variety of users including agencies of state government and other levels of government as well as the private sector. As a service organization, LRIS attempts to provide services as cost effectively as possible. The cost recovery nature of the work requires that LRIS look carefully at improving the means by which users are served. The most important issues involved in establishing a corporate database for LRIS are those which are related to providing a high quality of service at the lowest possible price for users.

The primary goals which LRIS hopes to attain in establishing a corporate database are to: (1) improve response to user demands for data and services; (2) maintain a high degree of data quality; (3) minimize extensive new data capture costs, a significant component of total GIS operating costs; and (4) develop better organization of data resources. Current and future user needs are the cornerstone for the design of a corporate database. Even with the best of intentions, a database designed and developed with the wrong data at the wrong resolution for the wrong users yields a data resource which will not be used as intended. A thorough examination of the characteristics of the LRIS user community is therefore necessary as a first step.

Data quality is a major issue as one begins to identify the components of a corporate database. The quality of service that LRIS provides starts with the quality of the data captured and used to perform spatial analyses. If those data are to be used successfully beyond a single, short term project, a high degree of quality must be assured. These data must also be available over the entire state at a level of resolution which is meaningful for analytical needs. LRIS has a substantial amount of data developed under the Comarc data structure. Cost/benefit assessments of quality, currency, and extent of coverage are parameters necessary to evaluate whether it is most beneficial to convert this data into a structure usable by ARC/INFO or to recapture or in some way obtain new data to replace the old Comarc data. These parameters, either individually or together, determine whether a particular project data set will be useful for conversion.

Data availability is another issue of concern. The corporate database is intended to represent statewide coverage for each layer of data selected for inclusion in the database. For some layers, data coverage is extremely incomplete and would thus require a substantial digitization investment. The best alternative would be to obtain digital data from other reliable sources to minimize the cost of data acquisition. LRIS has worked with the USGS, for example, to obtain digital data in a timely and efficient manner to fulfill some data needs.

In order to establish a corporate database and simultaneously meet the challenge that it entails, LRIS needs to ensure that the database contains data that are practical to maintain over the long term. This issue involves institutional considerations as much as, if not more than, technological ones. Data layers that are to be routinely maintained require the backing of the institutions that were the sources of those layers in the first place. The institutions must be committed to providing updates to the data on some reasonable schedule. Without this kind of support, such data become impractical for storage and maintenance in a corporate database.

DESIGN METHODOLOGY

LRIS has identified a methodology for designing the corporate database with the above goals and issues in mind. The design methodology being used for the corporate database consists of a nine-step process which is outlined below:

1. Evaluate current and future needs of the user community.
2. Inventory current data holdings.
3. Perform quality assessment of data holdings.
4. Assess data volume, currency, resolution, coverage, and frequency of use of data holdings.
5. Consider future application demands and derive data requirements to meet them.

6. Identify the most critical data to be stored and prioritize the remaining data for storage consideration.
7. Develop data organization strategy.
8. Prepare implementation plan.
9. Perform implementation of the database.

Each step produces information that is essential in achieving the overall design and implementation goals for the database.

Evaluate the current and future needs of the user community

This step identifies the application and data needs of current LRIS users and many potential users including such parameters as frequency of use of specific data layers; provides LRIS with insight into where finite resources should be spent and projects to target in the future.

Inventory current data holdings

LRIS has over 60 graphic layers of data and numerous associated tabular data sets in digital form which have been developed over the past 11 years for several hundred projects; simply documenting this data gives the organization a good start on the evaluation process.

Perform quality assessment of data holdings

This examination assists LRIS in determining whether specific project data sets are worthy of inclusion in a corporate database, which by definition will be maintained and updated on a routine basis; assessment of quality includes consideration of the source material and digitization methods used, among other factors.

Assess data volume, currency, resolution, coverage, and frequency of use of data holdings

These factors are useful in determining the utility of existing data holdings. Data volume provides a first indication of the potential size of the corporate database; data currency addresses the vintage of the data which is potentially important depending on the type of data; data resolution is concerned with the level of detail at which data were captured and indicates its possible usefulness in the long term as well as the relative cost of capturing and maintaining it; data coverage is the geographic extent of the digital data (e.g., statewide, a specific watershed, individual counties); and the frequency of use is an indicator of the value of a data layer or data set and may have implications on the design of the database as a whole.

Consider future application demands and derive data requirements to meet them

This step is based upon the earlier evaluation of user needs and in light of the results of the assessment of current data holdings. The product of this step is new data needs that should be accounted for in the corporate database design.

Identify the most critical data to be stored and prioritize the remaining data for storage consideration

This step considers the needs of the user community such as frequency of use of certain data and other parameters to arrive at a set of data storage priorities; it will also provide an indication of the data volume necessary to support users when coupled with size information for current data holdings.

Develop data organization strategy

This is essentially the detailed design portion of the methodology. Alternative strategies are derived which are workable in the LRIS organizational environment with the design goals enumerated earlier in mind. The strategy that best meets user needs and is acceptable to the LRIS director, production management, system management, project development, and database administration units of LRIS is selected for implementation. The key selection factors are: costs, impacts, and potential benefits.

Prepare implementation plan

This step lays out the schedule for implementing the corporate database. The plan includes not only the details of the process and the order of implementation but also time and resource estimates for completing the implementation.

Perform implementation of the database

Based on the plan, this step is the actual setup and loading of the corporate database in accordance with the schedule outlined.

A cyclical process is implied in the methodology whereby data needs and data holdings are continually being matched against one another and refined until a clear picture of the content and design of the corporate database emerges.

PROGRESS TO DATE

Progress toward development of an LRIS corporate database had been slow until recently due in part to the cost recovery nature of operations.

However, LRIS has been selected as the data management center for the Albemarle-Pamlico Estuarine Study (APES) Program, one of the National Estuarine Programs funded by the Environmental Protection Agency. The APES program is an appropriate vehicle for beginning to develop the corporate database for three reasons: (1) the APES study area includes the entire eastern portion of North Carolina, (2) LRIS is responsible for designing, implementing, and maintaining a GIS database for the study area based on a user needs assessment, and (3) funding is available to perform a thorough, formal database design.

At the time of this writing, the first three steps are nearing completion. The user needs evaluation consisted of a set of over 50 interviews with current and future user organizations to determine both application and data needs. Information from these interviews is being analyzed. The LRIS staff has concurrently developed an inventory of data holdings based on several hundred projects completed since 1977. These data are currently being assessed for their utility as part of the corporate database using data quality and user needs as primary criteria. The remaining steps in the methodology are scheduled to be completed during the 1989 calendar year, including actual implementation of the database.

CONCLUSIONS

The lessons that LRIS has learned in dealing with GIS project data sets over the past 11 years have been numerous. The design of a corporate database should have been completed sooner, but the cost recovery mode of LRIS operations kept the focus on day-to-day and week-to-week activities rather than the long term. Additionally, based on the experience of the organization in operating a GIS, a broader view of data has emerged in terms of how the data captured fits into an overall database scheme --- i.e., a corporate database.

LRIS expects to attain its goal of implementing a thoroughly designed, useful corporate database to take the organization into the 1990s in service to the user community. The reasons for anticipating this success are the solid methodology identified for achieving the goal, the APES program as a vehicle for boosting LRIS in that direction, and simply the fact that the time has come for the organization to implement such a database. The benefits are numerous and are directly related to providing a higher level of service in terms of cost, timeliness, and efficiency.

REFERENCES

Tribble, Thomas N., and Karen C. Siderelis. 1988. "Status of the State Geographic Information System in North Carolina". Proceedings of the Urban and Regional Information Systems Association, Los Angeles, Vol I: 48-57.

ISBN 0-944426-55-7