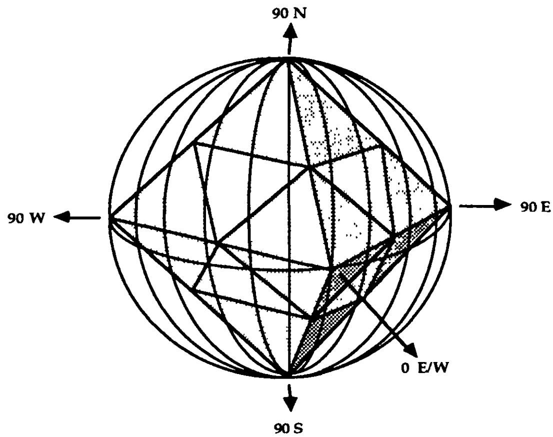
Planetary Modelling via Hierarchical Tessellation[1]

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 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 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.

[1]Based 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 *quadtrees*, 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 ($=4^{level}$), and column 3 ($=2^{level}$), 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

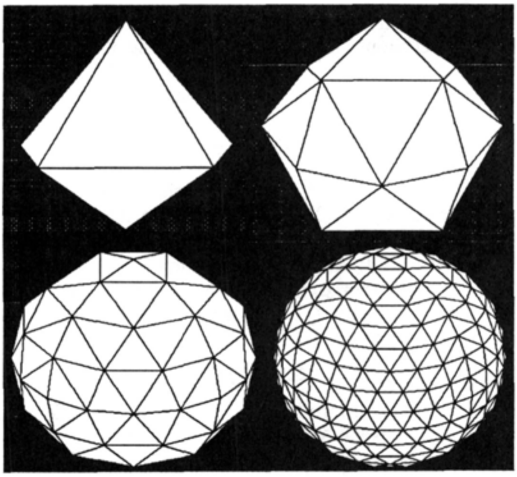


Figure 1 (left): Development of Quaternary Triangular Mesh to level 3 on a basis octahedron

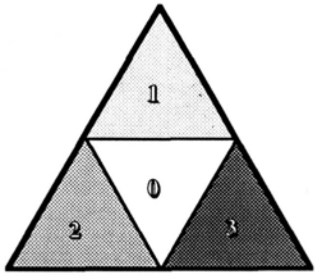


Figure 2: QTM Facet Numbering

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).

<u>*QTM as geocoding*</u>. 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 *3526B1CE*. 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).

<u>*QTM as geometry*</u>. To identify exactly where on earth *QTM* hex geocode *3526B1CE* (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 (0º N/S) and the Greenwich Meridian (0º E/W), and another (with its four surrounding tiles) at the antipode (180º E/W). Finally, each of the points where the equator intersects longitudes 90º E and 90º W colocate 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.

<u>*QTM as addressing*</u>. 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 quadtrees, they have the property that numerically similar *QTM* geocodes tend to lie in close spatial proximity to

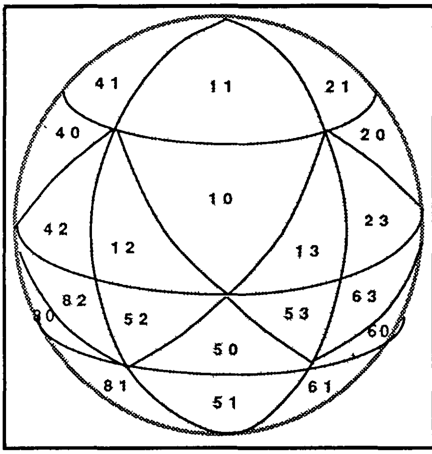


Figure 3a: First-order QTM tessellation of a sphere, showing facet numbering

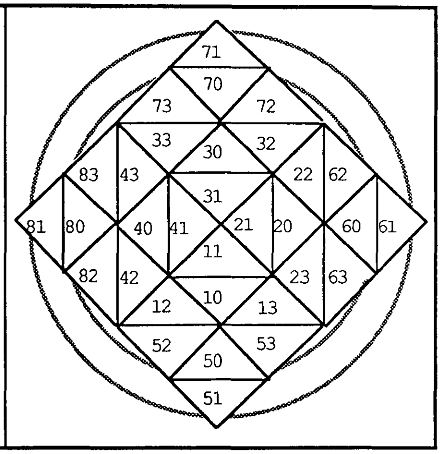


Figure 3b: First-order QTM tessellation of an octahedron, unfolded from S pole

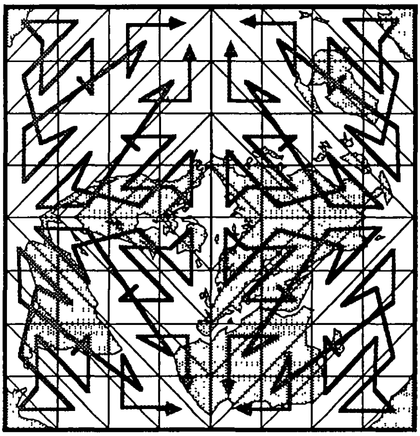


Figure 4: Second-order QTM Code Sequencing (memory map order). ZOT projection.

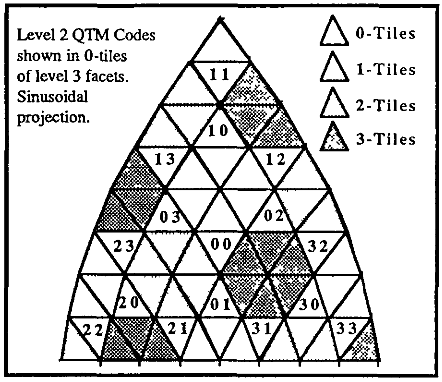


Figure 5: Pattern of least significant digits of QTM codes, forming hexagonal clusters (attractors)

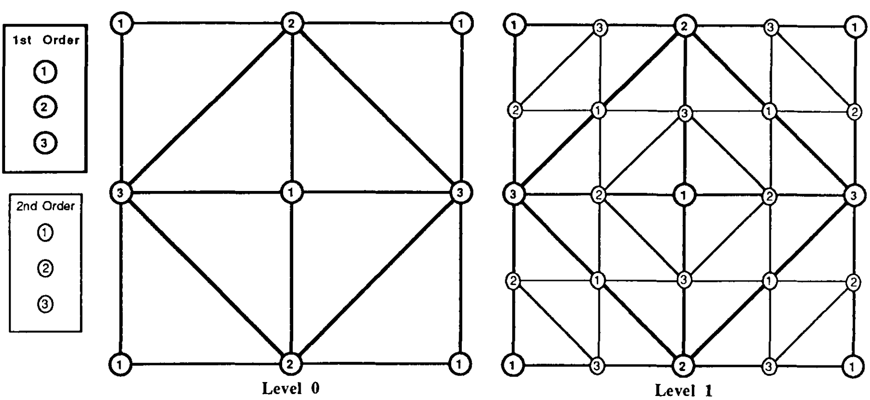


Figure 6: Octa and first level QTM Attractor (node) numerology; Child nodes are numbered 6 - (a+b), where a and b are the nodes of the parent edge.

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 Caucuses 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 tesselations; 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 tesselar 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, tesseral 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 Tesseral 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 quatrees 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.