Database-stored Representations and Overrides, Supporting Automated Cartography with Human Creativity

Paul Hardy, Cory Eicher, Marc-Olivier Briat, Thierry Kressmann
ESRI Inc, Redlands, California, USA
[phardy,ceicher,mbriat,tkressmann]@esri.com

ABSTRACT

Geographic information systems centered on relational databases are a powerful and proven way to collect, store, and analyze geographic data. Such systems are also used to produce cartographic products including maps and mapping datasets. However, existing mapping systems built on GIS databases fail to fully leverage relational database technology, mainly because most systems store geographic information – geometry and attributes – in the relational database, but store map definition and symbolization information in separate files. Also, map symbolization is accomplished by applying rules that assign symbology to sets of categorized features, a system that is seen by many cartographers as being too restrictive in not allowing one to interactively change individual cartographic graphic representations.

This paper proposes a GIS-based cartographic production system where cartographic information is stored with GIS data in the relational database. A system whereby dynamic cartographic symbolization and geometry processing rules are stored in ESRI's geodatabase is described. The representation of features is achieved initially by applying these sets of rules without manual intervention. The system then affords the cartographer the freedom to intervene and override any representation. Such overrides can take the form of changes to symbolization as well as geometry.

1 INTRODUCTION

1.1 Context

The most common approach to production cartography in recent years has been to use a GIS to generate and store geographic information in a spatial database, but then transfer it to a graphics package or a specific map finishing system to carry out the representation and cartographic edits. While this has the advantage of using best-of-breed tools, there are a number of drawbacks. Having to export, then import, data to transfer it between packages is time consuming, and duplicating changes and updates in both environments is inefficient and expensive. This approach is further limited by not providing a WYSIWYG connection throughout the process, by making feature attributes unavailable during finishing, and by the need to maintain separate databases to make multiple products at different scales.

1.2 Cartographic representation

The conceptual basis for cartographic representation has been the subject of extensive academic analysis [MacEachren 1995], [Fairbairn et al 2001], but there has been continued difficulty in resolving the conflicting pressures of automation (rule-driven visualization) with those of cartographic clarity (freedom of expression). Giedre Beconyte in a recent paper on "Conceptual

Models for Cartographic Representation" states "*Other than in the simplest cases, it is impossible to limit cartographic design to a single set of rules at all; hence thematic mapping can hardly be subject to automated processing functions*" [Beconyte 2004]. This paper introduces a system which unifies the automation and freedom aspects and hence contradicts the above analysis.

1.3 Requirements

Many mapping agencies have a strategic goal of using a common database and common environment for all map publishing, and a 'capture once and use many times' ethos. This requires:

- A single software environment from capture to finishing
- Centered on database
- Supporting multiple representations for multiple products
- Extensible to handle generalization and incremental update as these advanced facilities mature

1.4 The challenge of database versus freedom

The introduction of a database at the heart of the map production process has many advantages that are covered in more detail in section 3.2 below. In particular it facilitates data sharing, process automation, and the handling of updates. Most cartographic production agencies have adopted a database environment because of these strengths. However, in the past, the database approach to cartography was criticized from two opposing fronts:

- Not being comprehensive enough; storing only geographic features in the database, with all associated information such as symbology and map marginalia in files outside the database.
- Being too restrictive; being bound by simplistic symbolization rules based on feature class properties or geographic attributes.

For some years ESRI has been researching the marriage of geography with cartography, the extension of the GIS database to handle map presentation and cartographic details, and the fusion of database with freedom [ESRI 2004]. New capabilities covering these areas are the subject of the rest of this paper.

1.5 Concepts

The solution to the above requirements and challenges necessitates the introduction of new concepts into the GIS/cartographic software:

- cartographic representations
- overrides
- representation editing tools

These concepts are described in detail in later sections, but the fundamental idea of this solution is to add minimal information to feature classes in a database to store representation rules and graphical overrides to individual features.

1.6 Simple workflow case

In the simplest case, an organization has existing GIS data in a feature class, and wants to use it to produce a high-quality cartographic product. In this case, they add a cartographic representation to the feature class (see Fig 1).

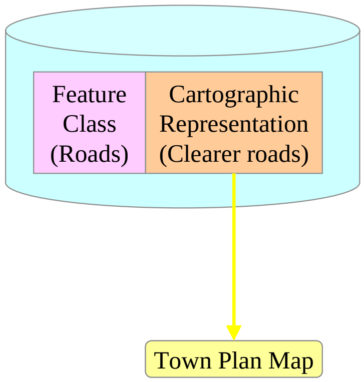

Fig 1. Simple case – Existing Feature Class

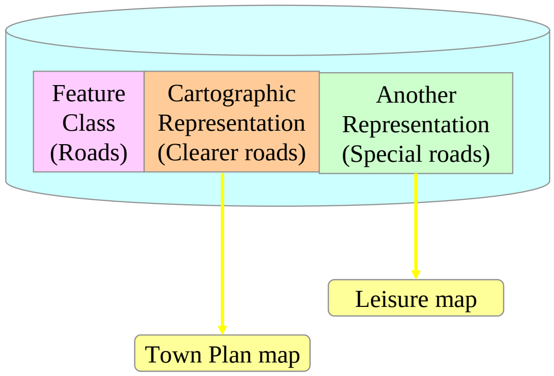

Fig 2. Multiple Representations

1.7 Multi-product Case

In the next case, an organization has existing GIS data in a feature class, and wants to use it to produce more than one high-quality cartographic product at similar scales. In this case, they add a cartographic representation to the feature class for each product (see Fig 2).

1.8 Enterprise Case

In the final case, an organization has a master landscape model data (DLM), and wants to use it to produce more than one high-quality cartographic product at different scales, as well as non-cartographic products (such as navigation routes for an in-car voice guidance system). In this case, they require the extraction of requisite data from the DLM by selection and generalization into a Digital Cartographic Model (DCM), which can then be enhanced with multiple representation capabilities as before (see Fig 3).

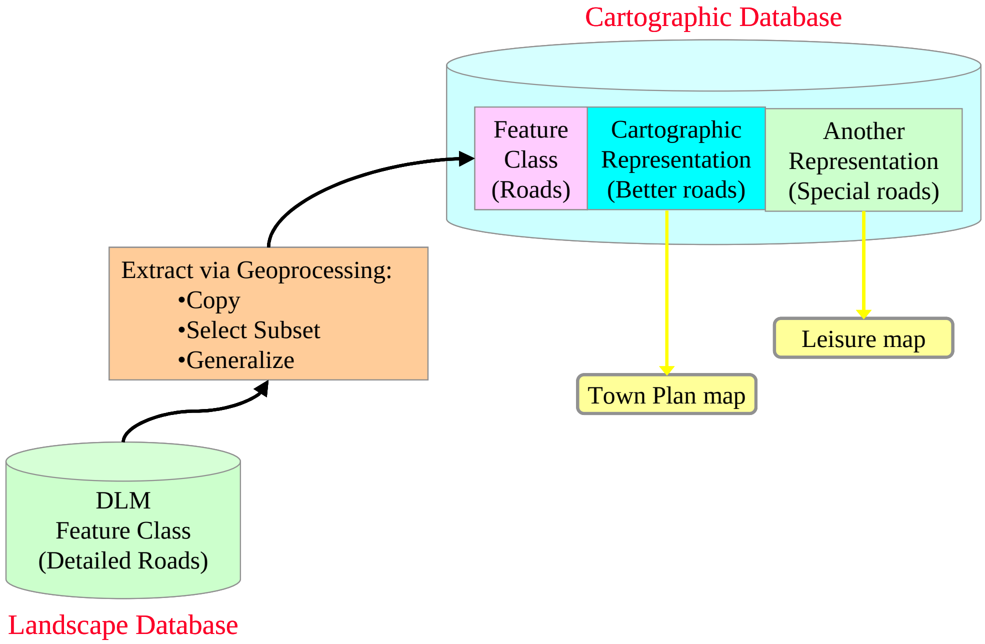

Fig 3. Enterprise case – DLM/DCM

2 EXAMPLES

2.1 Overview

The facilities of the proposed mapping system are capable of producing very high quality cartographic output across multiple domains. The following map examples demonstrate this.

2.2 Line example

For cartographers with discriminating eyes, representing linear features unambiguously with patterned or dashed lines has traditionally been a challenging problem. A database-centered mapping system provides a new solution to this and similar problems, by supporting high-quality automatic representation of GIS features, while at the same time providing the flexibility to override the automated rules. Clear and attractive maps can be efficiently produced.

The figures below show four stages of symbolization for linear road or ‘track’ features from a vector topographic GIS database (data courtesy swisstopo). Traditional GIS-based mapping systems support the first two symbolizations. In Figure 4 default GIS symbology is assigned to the linear features, and in Figure 5 a dashed line symbol is applied. This symbolization falls short of many cartographic requirements because the poorly symbolized line intersections and bends (both highlighted by the red circles), lead to ambiguity as to where the tracks start and end.

Fig. 4 Simple GIS Symbol

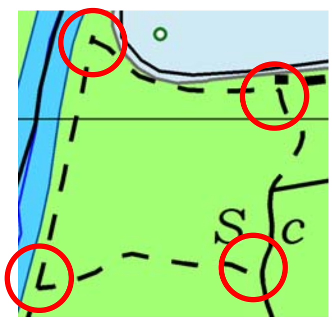

Fig. 5 Dashed GIS Symbol

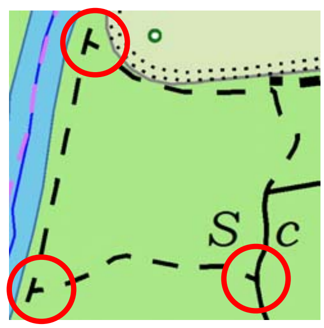

Fig. 6 Automatic Representation

Fig. 7 Manual Override

Figure 6 shows how the proposed mapping system can automatically produce better symbology at line intersections by adjusting the line dash pattern to ensure intentional connections at the ends of all such line features. Using this improved representation as a starting point, the cartographer can further perfect it using a manual override. To improve the representation of the sharp bend in the track in the northeast corner, the cartographer manually marks the corner as needing to be in the center of a dash. This modification is stored in the database as an override to the representation geometry.

2.3 Area example

In figures 8 through 11, increasing cartographic sophistication is shown for a polygonal cemetery. The first two states are achievable with a standard GIS, but a cartographer (or a pernickety map reader) may object to the depiction of partial grave markers.

Fig. 8 Simple GIS area

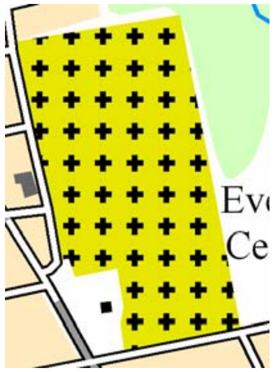

Fig. 9 – GIS patterned fill (note part symbols)

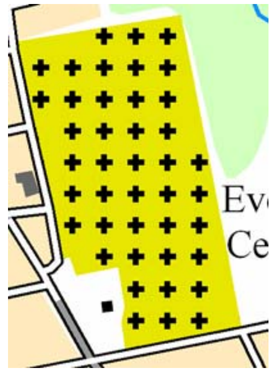

Fig. 10 – Rule-based representation

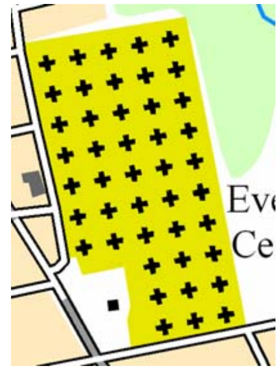

Fig. 11 – Overridden representation

Figure 10 introduces rule-based automatic representation, which ensures that only whole symbols are shown. Figure 11 shows the final clean cartographic effect, produced by overriding the angle of each symbol and the angle of the symbol grid itself. This could be accomplished either by human interactive override, or by a batch operation using the GIS toolkit to determine the predominant axis angle of all such cemeteries.

3 SOLUTION

3.1 Overview

This section describes in detail the proposed database centered mapping system. At the heart of ESRI's solution is the ArcGIS software product family [ESRI 2004B], with its geodatabase [ESRI 2004C]. The ESRI strategy of storing more cartographic information in the database is described in [Frye & Eicher 2003].

The functionality advance being described in this paper is that the database stores not only vector geographic data, including geometries and attributes, but also cartographic information that describes the representation of that data. By closely coupling the storage of cartographic representation information together with geographic information in the database, a new representation symbolization process is defined. During this process, vector GIS features are accessed from the database, symbolized, and output to the screen or printed map. The process introduces some new concepts including geometric filters that pre-process geometry before symbolization, and stored structures that override individual feature representations.

Overrides allow exceptions to the automatic representation rules stored in the database. They too are stored in the database, and can be used to represent a given feature or set of features differently than as defined by the rule for the feature set. An override can constitute a different symbol appearance, such as a thicker line symbol or a different fill color, or a modified geometry from the original feature geometry. Overrides can be created manually by the cartographer, or automatically in batch-fashion by running a geoprocessing operation.

Cartographic editing tools complete the mapping system. These interactive tools allow cartographers to modify feature representations to polish their maps. Tools alter both symbology and geometry, and store changes as overrides. Cartographers can interact with their map in a WYSIWYG environment.

The representation storage, overrides and tools were based on design concepts made available to ESRI when it acquired the French software company ALIDA and its DataDraw product [ESRI 2003].

The broader design of the proposed system, when scaled to enterprise solutions, can be based on a two-part database modeling of digital map information, where true-to-ground geographic information comprising a digital landscape model, or DLM, may be stored separately in the database from cartographic information, specific to a given map or map product. This cartographic information, plus map and page definitions, constitute a digital cartographic model, or DCM. Generalization, representation, and intervention are the primary processes linking such a DLM and DCM, and this paper concentrates on the representation and intervention stages.

3.2 A cartographic information system based on a relational geodatabase

Geographic information systems based on commercial software and relational database technology are widely accepted and have proven to be useful when applied to domains such as natural resources, real estate, law enforcement, public utilities, transportation, and many others. Cartographic production systems have been built on the same basic technology model, and these systems thus leverage the many benefits of database-centered information systems.

Using a relational geodatabase to store geographic and cartographic information provides the following advantages:

- the organizational advantage of a versioned database which allows many users to simultaneously edit data using long-transactions
- the ability to store geographic and cartographic information using standard RDBMS technology including Oracle®, Microsoft® SQL Server™, Informix®, and IBM® DB2®
- the ability to leverage DBMS management and administration personnel skill sets and information technology (IT) infrastructures
- conformance to corporate IT standards, and hence sharing of data across the enterprise
- data can be used not just for cartography, but also for analysis, on-screen views, and mission critical applications (see Fig. 12)

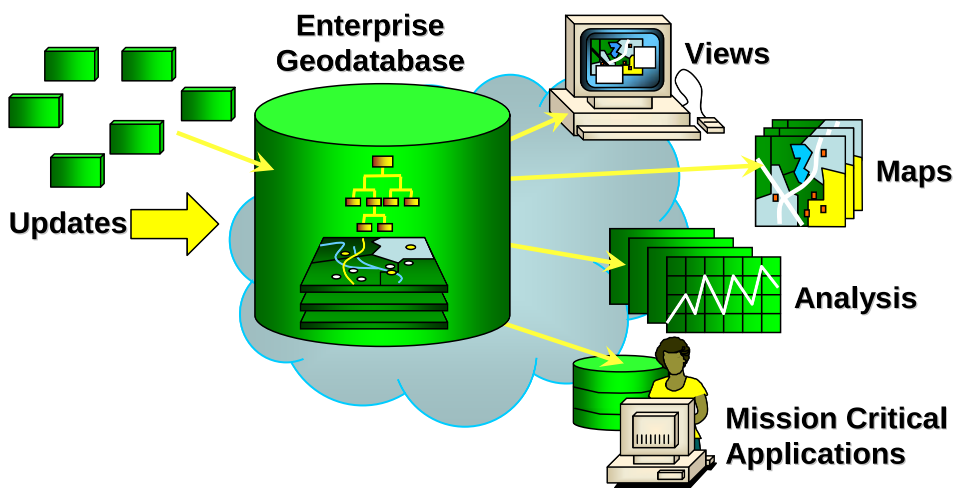

Fig. 12 Enterprise Geographic Information System

Additionally, a “single system” strategy for data storage and map production, based on a geodatabase and GIS technology, affords several benefits. Using the same system to collect and maintain GIS data as to produce and update maps and cartographic products:

- reduces data redundancy
- maintains access to GIS attributes from cartographic “linework”
- facilitates easier incremental update of maps when source data is updated
- avoids expensive rework, when correction has to be applied in multiple copies

3.3 Cartographic representations

A key concept that strengthens a GIS-based map production system is the ability to store cartographic representation information in the database, together with the geographic data being represented. Current out-of-the box, or so-called commercial off-the-shelf (COTS), facilities for this are limited in GIS software, but a new technology is being developed at ESRI for this.

A cartographic representation offers a superset of the capabilities of a GIS layer. One of a representation’s purposes is to encode information on how to categorize and symbolize features. Storing this representation information in the geodatabase contrasts with the setup that is most common in current systems, where layer information is typically stored in binary map document, or in separate layer files. Advantages of storing representation information in a relational database follow from the general advantages (listed above) of database-centered information systems.

Physically, a cartographic representation adds additional columns to a standard ArcGIS feature class table. As in any vector GIS, the feature class stores point, line, or polygon geometries, as well as a set of additional attribute columns used for mapping, analysis, and data management. The added representation columns store data that defines the representation rule used to symbolize a feature. They also store cartographic overrides, which are exceptions to the representation rule for a given feature.

For example, when representing a feature class containing roads, different representation rules are defined for each unique road class, such as highways or local streets. Overrides then allow a cartographer to modify an individual symbolized road feature, giving it a different appearance than the rule dictates, perhaps changing the width of the symbol of a local street from 5 mm to 3 mm. This operation creates an override that affects only the representation of this single road feature. Overrides are covered further in Section 3.5.

A basic system design premise was to avoid unnecessarily duplication of data to produce representations. Therefore, the extra columns that define a cartographic representation are minimal in size, and wherever possible the representation information is derived from the existing GIS feature as needed. The structures used to hold overrides are highly flexible, and avoid the need for separate columns for individual overrides.

As defined, a cartographic representation encodes information about the representation of a feature class. This means that a cartographic representation cannot exist without a feature class. A feature class can support multiple cartographic representations, and for each representation added to a feature class, additional information is added to the feature class table.

3.4 Representation Symbolization Process

The mechanism by which a *GIS feature*—one row of a regular GIS feature class—is symbolized for output to the screen or map is the *representation symbolization process*. See Figure 13 for a schematic diagram of this process.

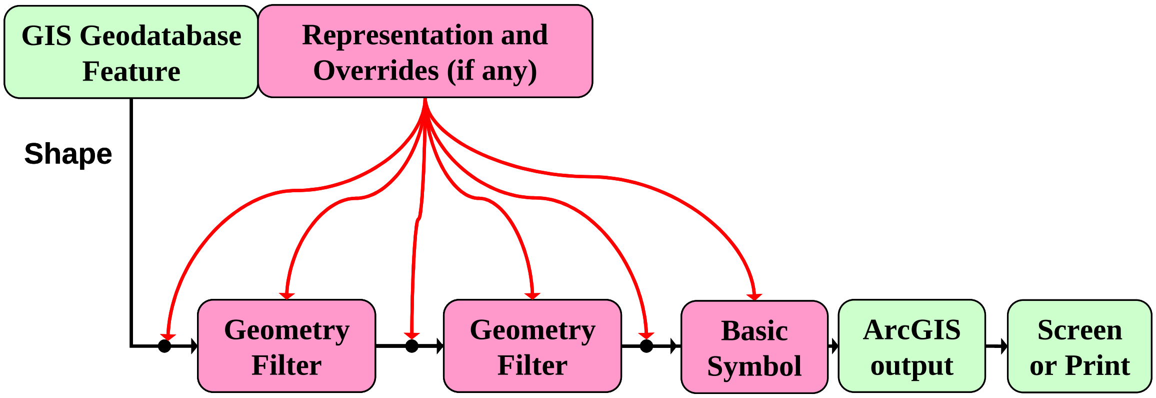

Fig. 13 Schematic Diagram of Representation Symbolization Process

The process begins with the feature geometry stored in the shape field of the GIS feature. In a traditional system—one without cartographic representations—this geometry is simply assigned a symbol and drawn on the screen. Surely, different symbols can be assigned to different features, and special logic can be used to assign symbols to features, as in classification logic used to create a choropleth map, but this is the basic process. A *symbol* defines how to draw a given geometry, for example a line symbol assigns color, line thickness, and line cap and join properties. In the ArcGIS implementation, a symbol also defines rules for producing different graphical effects, such as producing dashed lines with specific dash length and dash spacing.

With the introduction of cartographic representations, the feature drawing process is different than in a traditional vector GIS: it produces higher quality symbolizations, and it is more flexible for the cartographer. To produce a cartographic representation, a *basic symbol* (see Fig. 13) is assigned to the feature geometry and the feature representation is drawn on the screen. A basic symbol is a software component that characterizes how to draw a point, line, or polygon geometry, with a limited set of parameters. For example, a basic line symbol would assign a color, thickness, and cap and join properties, but would not have options for dashed lines.

Still referring to Figure 13, an additional component called a *geometric filter* can be inserted in the representation symbolization process. The purpose of a geometric filter is to pre-process geometry before a basic symbol is assigned and before the feature is drawn. Cartographers can choose from many filters. For example, a filter can be inserted to produce dashed lines, or another filter can be inserted to offset line geometry by a given distance. Multiple filters can be inserted together in series; when they are, the filters are said to be *chained*. Users can write their own filters (to do customer-specific tasks) and link them into the process.

Additionally, filters can modify the type of geometry of the features. For example, a filter could be added to accept polygon geometry as input, and output a single point for each polygon at the polygon centroid. Filters have their own set of configurable parameters, just as basic symbols do. For example, a filter that produces dashed lines has parameters that let the cartographer specify dash lengths, gap lengths, and the ability to control special effects at line ends or bends.

Similarly, a filter that processes and offsets line geometry has a parameter to specify the line offset in either page or map units.

3.5 Overrides

Some professional cartographers have traditionally avoided working with GIS-based mapping systems and software, especially in the final polishing stages of production work. They have preferred to work in file-based graphic software packages, either generic graphics packages or software more focused on cartography. As cited by these potential users, the main limitation of database-centered GIS mapping systems is representation inflexibility. In traditional GIS-based mapping systems cartographers have not been able to modify graphical and geometric properties of individual feature representations. The proposed mapping system and representation symbolization process being developed by ESRI provides this flexibility by introducing the concept of overrides.

Overrides were introduced in an earlier section as an exception to the representation symbolization that is defined in the representation rule. Just as a cartographer working in a graphic software package can select a polygon graphic and change its fill color and outline thickness, the cartographer can now perform the same operation in a database-centric, GIS mapping system. The changed thickness and color properties are the overrides. For another example, consider a map where a cartographer wants to override the ‘line cap’ (round or square end) symbol property of a dead-end road. They might also modify the geometry of the road to resolve a conflict with a nearby river. Both these changes can be stored as overrides.

Overrides are stored in representation fields added to a GIS feature class. The system is structured to support multiple overrides for a given feature representation. Continuing with the previous example, this means that the cartographer can override a road’s line cap style and its geometry and have that information stored with the feature in the geodatabase.

The system implementation handles overrides in a very generic way. Cartographers are able to override basic symbol properties, geometric filter parameters, as well as feature geometry, and even the parameters of custom developed filters.

Overriding feature geometry can be useful if one wants to modify a feature shape for cartographic purposes, while keeping the original feature geometry intact. This method of storage keeps the original shape geometry, for use in analysis or in another cartographic product. Also, as previously mentioned, the system supports multiple cartographic representations for a feature class. This possibility, coupled with a cartographic representation’s native support for storing an alternate representation (via feature geometry override), means that the system can support powerful models where the same feature can be represented differently in multiple map products. For example, when producing a topographic map and a transportation map, each map can have its own representation of the same feature, and these representations can have different geometries.

3.6 Cartographic Editing Tools

Together with overrides, the introduction of cartographic representation editing tools allows cartographers who currently prefer to work using graphic-centric software to work even more efficiently, in a GIS-based system. Representation editing tools that work similarly to tools found in graphic packages such as Adobe Illustrator or Macromedia Freehand are introduced to

ArcGIS. Furthermore, many of the newly introduced tools are more efficient because they are designed with specific cartographic tasks in mind. For example, tools are provided to orient markers—or sets of markers—to adjacent linear features. Other tools allow the cartographer to produce line work that is offset cleanly from the original.

The editing tools provided by the system (see Figure 12) are divided into three main categories: tools for editing geometry, tools for editing attributes, and tools that allow you to create and modify feature representations based on the interaction of one representation with another. Editing of representations takes place within the same versioned editing environment currently supported by ArcGIS for editing vector feature classes.

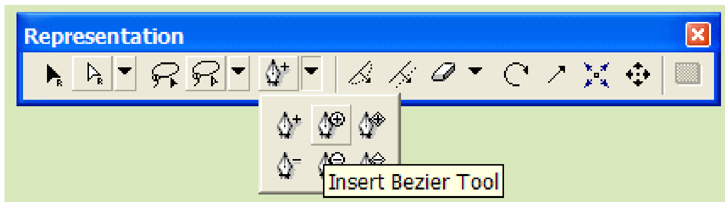

Fig. 14 Cartographic Editing Tools

Representation attribute and geometry editing begin with two basic selection and modification tools whose behavior will be familiar to cartographers experienced with graphics software (see Fig. 14, left side). The representation editing tool, called the black tool, selects whole feature representations. The geometry editing tool, called the white tool, selects parts of one or more representation geometries.

A number of behaviors are built into the system to facilitate the work of the cartographer when editing (see Figure 14). When selecting representations, selection is based on the symbolized representation, not just the geometry. In other words, a cartographer can select a line representation by clicking anywhere on the visible line; it is not necessary to find and be within a certain tolerance of the centerline.

3.6.1 Geometry editing

A nice feature when editing geometry is the ability to see linework and vertices without changing to a special "sketch mode". Also, to again emphasize the contrast with traditional GIS feature editing, both primary editing tools can modify multiple feature representations at once, including their geometry. In fact, parts of the geometries of multiple representations can be modified simultaneously using the geometry editing tool. For example, a cartographer may select parts of three intersecting linear roads whose representations intersect at a ‘T’ and then reposition the selected segments of these three geometries, including the intersection itself, in a single edit.

When working with linear and polygon geometries, cartographers can interact with segment geometry and modify and manage vertices. Efficient tools are provided for adding and removing vertices, as well as for changing the status of a vertex from a simple vertex, to a special vertex that manages one or more Bezier curve segments. Intuitive tools are provided for working with Bezier curves that are familiar to users of existing graphic software packages. Additional geometry editing tools are provided that allow a cartographer to work with masks, to erase geometry or parts of geometry, and to shift and warp representations.

Using these geometry editing tools allows the cartographer modify the representation to obtain the necessary product clarity and elegance, without breaking the GIS data and its ‘true-to-ground' relationships. This is particularly important when updates have to occur as change happens in the real world.

3.6.2 Attribute editing

A second main class of editing tools is provided for working with attributes. As mentioned, the black representation editing tool is the tool used for selecting and modifying one or more whole feature representations.

A given feature representation has only one set of graphical attributes, and the structure of these attributes is defined by the representation's representation rule. As an example of how a cartographer can edit representation attributes, consider capital cities represented by a rule consisting of a star-shaped marker, with a color attribute, and a specified symbol size. These structures are fixed for the representations (capital cities) to which these representation rules are assigned, but the values of these attributes can be changed on a feature representation by feature representation basis, using the graphical attributes palette. So, a cartographer can select one capital city representation (the national capital) with the black representation editing tool and change its color to distinguish it from the provincial capitals.

Cartographers have other ways to modify graphical attributes, too. An interactive tool is provided that exposes graphic attributes for modification. A simple example is the modification of marker size. The cartographer selects the national capital point representation and can modify it by clicking and dragging with the mouse to change the marker size interactively on the map.

To edit both the geometry and the attributes of representations, there exists a set of tools that create or modify representations based on the interaction between two representations. In the proposed system, a generic drag-and-drop tool is provided. A cartographer can select this tool and then specify a behavior, such as masking. To invoke, one representation is dragged and dropped on another representation using the mouse. This operation masks the second feature with the first, creating a visually clear area around their intersection. Other drag-and-drop behaviors are defined by the system for creating and modifying bridge-tunnel representations where two lines cross, as well as other common cartographic effects. The system can be extended by adding user-written custom drag-and-drop behaviors to accomplish specific editing tasks.

4 CONCLUSIONS

Advanced cartographic representation mechanisms and tools similar to those described above will soon become available in a commodity GIS, and will dramatically change the scope of automated cartography, facilitating generation of multiple products from a central database. The human freedom enabled by the override system, combined with the symbolization rule pipeline, will permit high-quality attractive cartography within a database-centered environment.

These mechanisms and tools are part of an overall vision to provide a single, consistent, intuitive, efficient, and liberating mapping environment, used throughout the cartographic communication process. This environment is geodatabase-centered, holding master data, plus specifications, processes and results of derived products. It releases cartographers from drudgery of repetitive actions, but provides freedom and tools to amplify our human creative and expressive skills.

5 NOTES

1. The sample data used in figures 4 to 7 is swisstopo VECTOR25, copyright Swiss Federal Office of Topography. The data used in figures 8 to 11 is courtesy of USGS.
2. This paper is a forward-looking document, and the capabilities it describes are still under development. As such, it is intended to give guidance as to likely future direction and should not be interpreted as a commitment by ESRI to provide precise capabilities in specific releases.

6 REFERENCES

- Beconyte G 2004, “Conceptual Models for Cartographic Representation”, Geovizualization 2004, Kolymbari, Crete http://www.kc.gf.vu.lt/Publications/ConcptModelling2004.doc

- ESRI 2003, ESRI Press Release, “ESRI Acquires DataDraw Firm, ALIDA”, February 5, 2003, http://www.esri.com/news/releases/03_1qtr/alida.html

- ESRI 2004, ESRI White Paper, “Cartographic Design Process – Artistic Interpretation With The Geodatabase”, http://www.esri.com/library/whitepapers/pdfs/cartographic-design.pdf

- ESRI 2004B, “What is ArcGIS”, ESRI Press, ISBN 1-58948-090-2, 2004, http://store.esri.com/esri/showdetl.cfm?SID=2&Product_ID=109&Category_ID=28

- ESRI 2004C, “Building a Geodatabase”, ESRI Press, 2004, http://store.esri.com/esri/showdetl.cfm?SID=2&Product_ID=90&Category_ID=28

- Fairbairn D., G. Andrienko, N. Andrienko, G. Buziek, J. Dykes, 2001, “Representation and its relationship with cartographic visualization: a research agenda” - Cartography and Geographic Information Science", Vol. 28, Nr. 1

- Frye C & Eicher C 2003, “Modeling Active Database-Driven Cartography within GIS Databases”, 21st ICA Conference Proceedings, Durban, S. Africa

- MacEachren A. M. 1995, “How Maps Work. Representation, Visualization and Design”. New York-London: The Guilford Press.