Cambridge Conference 2003

www.cambridgeconference2003.com

Ordnance Survey
Romsey Road
SOUTHAMPTON
SO16 4GU
United Kingdom

Intelligent Multi-Scale Cartographic Data and Their Databases

C E Frye
ESRI
USA

Paper 4C.1

Ordnance Survey

Intelligent Multi-Scale Cartographic Data and Their Databases

C E Frye
ESRI
USA

Summary

In a world overrun with information and consumers of digital information expect technology to make raw data presentable in seconds and finished content in hours. In this context, map makers are behind the times. The content and the processes for making commercially viable and government mandated maps are complex and must address using terabytes of imagery and detailed spatial data. Currently, many such maps require large amounts of budget-draining human attention to complete. Of the software that is prevalently used for making maps, none is capable of rapidly producing these maps. Rapidly, in this instance is not relative to how fast these maps were made ten and fifty years ago, but rather rapidly relative to how other vast quantities of data are processed in databases and by production software. For instance, the databases behind the world's banks and stock exchanges are capable of and are now expected to perform complex tasks on demand. In fairness to map makers, the irresistible forces of market-driven economies drove the speed and maturity of these financial databases and software systems. Thus, this paper's purpose is to describe an evolution in the maturity of the databases and software systems for map making.

Currently a smart database that contains everything a mapping software application needs to a finish publication quality map with a demanding cartographic specification without human intervention is much hoped for, but does not yet exist. To create a database that can continuously drive a mapping software application to produce up to date maps at any hour of any day, a well designed database is needed. Research on such a design at ESRI is beginning to show that using traditional database objects like tables and relationships can encompass the complexities of cartographic information and the processes that produce high quality maps.

Even with a well designed database, a mapping software system that can intelligently follow a production process without intervention is needed. Furthermore, the manner that a user of such software defines the map and the process to build it should not be reliant on an overly technical solution. Users should be able to pick functions, choose options for those functions without writing source code and be able to automate the processing of their data to produce a map without writing source code. With the next release of ArcGIS, such a system will be available. The geoprocessing framework that will be integrated into ArcGIS fulfills these requirements and with very little technical overhead. A cartographer will be able to design a map-making process by dragging and dropping data objects and data processing functions into a model. Furthermore, such a model can be stored, shared and managed in any of the industry standard database management systems (DBMS) that ArcGIS supports.

In the future, intelligent cartography will be the process and result of a robust software system creating cartographic products from well-modeled data. The two main points made above are that a sound database design is needed and that maps can be modeled as processes are seminal to ESRI's research efforts. To create a software system for intelligent cartography, a robust articulation of a cartographic model is needed, along with a well designed database model for holding this cartographic model. This paper will describe both and generally, how such a software system should work.

A Basis for Intelligent Cartography

Intelligent cartography is the process and result of a robust software system creating cartographic products from well-modeled data. One of ESRI's goals is to build such a software system. However, terms like "robust", "system", and "well-modeled" are often used and not explained, leaving wide berth for interpretation. In this context, these terms are explained as follows:

- **Robust:** Robust software in a GIS context means that the software functionality has been designed and has been tested for an exhaustive gamut of usage scenarios. In particular, it can be said that software can be implemented such that it performs its designed tasks on one set of data very well, but robust software can perform those tasks on any set of data as long as the same minimum criteria can be used to describe the data.
- **System:** As a system, GIS software provides tools that allow organizations and users to create, manage, analyze, and visualize geographic information, Figure 1. The "system" aspect in this case has to do with the fact that the work of these organizations and users has a purpose. That purpose is defined as the business of that organization. This paper will describe a system that is relevant to persons and organizations that make maps.

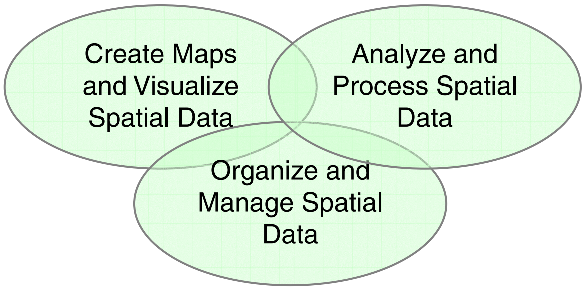

Figure 1. Three tasks that GIS software facilitates, automates

- **Well-Modeled:** A description of well-modeled GIS data includes the ideas that the data are consistently categorized with respect to a specification. This allows processes to be defined that use the data and then rules to be defined that validate the results of the processes. In other words, well-modeled GIS data can be processed into cartographic representations that conform to and are validated by a set of rules.

It is no accident that Figure 1 depicts analysis and visualization atop a strong foundation of well-modeled data. Well-modeled data are the basis for any effective information system. The software tools that work on the data are then the means for any organization to successfully use their data to accomplish their business goals. This paper will describe a conceptual data model [Nicewarner, 2003] for a well-modeled multi-purpose Geodatabase [Zeiler, 1999] and a set software processes for ArcGIS to intelligently produce several different kinds of maps. The intention is this approach can be adopted by any mapping organization to produce their maps more efficiently and intelligently than ever before.

In order to have a well-designed database to support intelligent cartography a robust description of the specification for each of the maps that will be produced from the single multi-purpose set of GIS data is needed. The idea is that the condition and characteristics of the GIS data must be known and the cartographic requirements for any product must be known before a data model can be designed. This is the most important aspect behind this approach to intelligent cartography: a map is a holistic context and is the basis for making decisions about how to automate intelligent cartographic production.

A term that will be used throughout this paper is cartographic model; a cartographic model is the combination of a symbol specification and a page layout specification. As stated earlier, a complete understanding of the cartographic model for each kind of map that will be produced from a given geodatabase must be fully articulated in a way that can provide a basis for a Geodatabase design and rules that govern the behavior of the data modeled using that design. To date, there is no widely accepted or consistent manner for depicting or storing a cartographic model. For the past two years ESRI has researched and published a series of domain-specific standard Geodatabase models to help its customers learn how to model their specific kinds of data. One of these models is for topographic base maps [Frye, 2003], and this model includes a notation for describing a cartographic model in conjunction with a GIS data model. In this data model, everything a map maker needs is described, including details about what information needs to be added to a GIS data model to support cartographic production.

One predicate of this paper is that an ArcGIS and standard relational database technology is the basis for this proposed framework for intelligent automated cartography. As such, there are two sets of assumptions that will fundamentally influence the discussion of how the database and cartographic models are used within ArcGIS software applications and the Geodatabase. First are the assumptions about the roles of the database and the software applications. Second are the assumptions about the character of spatial data that are stored in the Geodatabase. For additional information about Geodatabases see [Zeiler, 1999].

- **Relationship Between Software and Database:** In ArcGIS the Geodatabase stores spatial data in tables. The Geodatabase also defines behaviors for that data. In the database these behaviors are the basis for managing the integrity of geographic data. The ArcGIS software applications provide the functionality that create, edit, analyze and display this geographic data. The Geodatabase is effectively a logical repository for information that the software applications produce and work with.
- **Character of Spatial Data within the Geodatabase:** Geographic data stored in the Geodatabase can be modeled in almost any traditional manner that GIS data are stored; tiles, study areas and most importantly as seamless continuous sets of thematically consistent classes of information. Furthermore, the Geodatabase provides the basis for a versioned, multi-user data production, editing and management environment.

This paper will at a conceptual level describe the nature and form of a cartographic and database model that can be used as a system to realize intelligent automated cartography. The role of the mapping software will be woven into these descriptions. In addition, discussion of more traditional cartographic concepts: symbology, text, generalization, etc. will occur throughout.

The Cartographic Model

There are several types of cartographic information contained in a cartographic model. These are really more specific information models for various cartographic aspects of a map. These are defined as follows:

- **Semantic Model:** The terms used to describe the features for a given map, for example is a water body with an area greater than 10 km^2 a “Lake” or an “Inland Water Body”; furthermore, does it need to be qualified in terms of its salinity or perenniality? Much of the semantic model should be contained in database domains, which are effectively look-up tables for feature type codes.
- **Symbology Model:** Describes how GIS features are represented on a given kind of map. Representation includes the specification for how features are drawn, which in turn includes how and where text, if needed to enhance the feature’s meaning, is drawn.

- **Cartographic Topology Model:** Just like GIS data need topological rules to govern the nature of their spatial interrelationships [see Hoel, et. al., 2003], symbolized representations of those data also have topological rules for governing the nature of their representational integrity. Specifically, the symbolized representations of GIS data have specific rules for interaction for each kind of style and scale of map.
- **Tools/Processes[1]:** Each layer of information in a map is derived from a GIS feature class in the Geodatabase; a feature class can be thought of as a layer of thematically consistent spatial data that is stored in a table in a database where each row represents a distinct feature. Often, simply assigning a symbol or drawing method to the geometry of each feature for the context of a given map is not enough to properly represent that feature in that map. To accommodate this, software tools can be used to process the data into an appropriate representation. These tools may change the spatial representation, i.e., the geometry, they may filter which features are shown on a map, and they may reclassify features into different, more appropriate representational categories. Regardless of the function of the tools, they must be designed to accommodate the context of the map, in particular follow the rules of the cartographic topology model. To make intelligent cartography automated, these tools must also be stored and managed in a way that permits them to be sequenced into a process model for producing the map.
- **Page Layout Specification:** The arrangement of the elements of a map on the page. This includes rules for elements that may be dynamic, based either on which sheet or page they appear on, or on the locale depicted by the map. These dynamic elements are also important for automated intelligent cartography in that they allow human intervention, and some kinds of human error to be eliminated from the cartographic production process.

To support mapping organizations in efficiently producing and managing cartographic information, it is necessary to store all cartographic information in the Geodatabase. This allows anybody in a given organization to have access to the data. Currently functionality to store all of these kinds of cartographic information in the Geodatabase is being implemented at ESRI for a future release of ArcGIS. Figure 2 shows a diagram that represents a very high-level conceptual model of a geodatabase that contains a cartographic model.

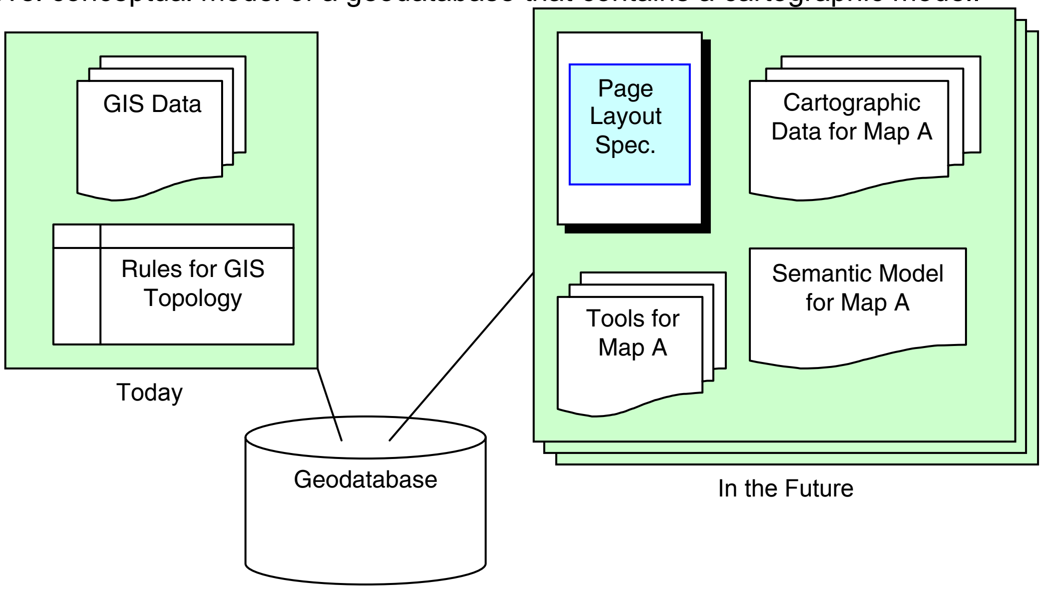

Figure 2. Conceptual Storage Model of a Geodatabase with Cartographic Data

[1] Note that the author feels strongly that the term, "generalization" is has too many meanings and as such will cannot be used in this paper. However many of the tools and processes mentioned here are what many people would consider as some aspect of "generalization".

Also shown in Figure 2 is the idea that one set of GIS data can be used as the basis for many different maps. The Geodatabase will store a cartographic model for each type of map. Not shown in Figure 2 is the idea that each specific instance of a map is also stored in the Geodatabase. Each map is managed such that it knows which cartographic model it uses. Each map is really a collection of elements and layers that each derive cartographic properties from the cartographic model.

Another important aspect of a system for automated intelligent cartography is the mapping software application. In ArcGIS this is ArcMap. ArcMap, in terms of what Figure 2 depicts, is responsible for creating and managing cartographic properties on specific elements and layers that are contained in maps. ArcMap also is where automated processes are prototyped, created and executed.

Conceptual Geodatabase Model with Cartographic Information

The cartographic model contains all types of cartographic information needed to complete a map. The form of these pieces of information is important and must be succinct, accessible and manageable. A conceptual data model is the set of objects and their logical roles [Nicewarner, 2003] that will ultimately have a physical manifestation in the database . This section will present the major types of cartographic data objects that will support intelligent cartography and describe how these objects will need to interrelate.

In order for cartographic objects to exhibit intelligence, they must have access to one another. By storing these objects in a database in a systematic way, that access will be implicit. Furthermore, this intelligence will be driven and governed by rules that describe the relationships between, and the storage parameters for cartographic objects. This gives the mapping software a means for discovering and working with the cartographic objects in a meaningful way. Table 1 shows the major kinds of information that are part of the conceptual database model for the smart map and Table 2 describes some of the key behaviors of those objects.

Table 1.

Object	**Description**
Map Dataset	A map dataset is a generic organizing object. A map dataset contains everything for the cartographic model for a given map or map series. The main idea is that all cartographic data for all pages use the same general cartographic specification are contained in the project. This allows for sensible breakdowns in assignments and database management of users relative to their work assignments. A map dataset is not necessarily a physical storage model, but rather a conceptual or referential model. Hypothetically, each kind of cartographic object could be stored in a table in the database where each table represents a different kind of object and each row represents a record. Map datasets would also be in a table, and their row identifier would be appear in a field in the other objects table so they would know which map dataset they were used in. This was obviously a simplified example as a well organized database would have normalized these objects such that a particular object could be used with any of several maps or map datasets.
Maps	In the context of this paper a map is the sum of what is drawn on a page. Typically some geographic data will be drawn using some symbols of a specific design with supporting cartographic marginalia constitute a map.

Object	**Description**
Page Layout	This is the definition of the page in terms of how large, its units, and potentially any printer specific information needed to restrict the placement of information on the page. The page layout is also the specification of which graphical elements, e.g., marginalia that will appear on the page, including any rules that govern the positioning of these elements relative to either the page or each other. The page layout also contains a collection of object relationships that describe how objects on the page relate to each other. For example, a graphical scale bar relates to a data frame and the scale bar will update certain of its properties each time the scale of the data frame is changed. Other kinds of information relationships besides those between two cartographic objects exist on the page layout. In particular there may be a relationship between the computer operator or more generically the computer and some text on a map. Thus work on each object can be captured, displayed, or managed based on who or which computer was used to do work on the map.
Element Collection	The element collection is the set of map elements (north arrows, scale bars, text, neat lines and data frames, etc.) that are drawn on the map.
Layers	Layers are the set of instructions for drawing a particular set of GIS or Cartographic data on a map. Each map has a collection of layers that are drawn within the data frames on a given page layout.
Symbol Collection Cartographic Feature Class	These are predefined symbols and parts of symbols, like colors, that are used by the mapping software to draw the map. Symbol collections are used in two contexts. First is that the objects in the symbol collection are used by the elements in a map's elements collection. The objects in the symbol collection are referenced such that each element know which symbols it used and each symbol knows which elements depend on it. The second use context for the symbol collection is in a specialized cartographic feature class. A cartographic feature class is derived from the combination of a layer and a GIS feature class. The cartographic contains a minimal distillation of the original data. The value of a cartographic feature class is that it can contain alternative representations of features that vary any aspect of how that feature is represented on a given map. Cartographic feature classes can store, as part of these alternate representations additional geometry and references to specific symbols to use with that geometry in order to represent features. Thus, a symbol collection, like with elements must reference specific features in a cartographic feature class and vice versa.

Object	**Description**
Location- and Sheet-specific Indexes	These indexes provide display properties for the map depending on the geographic extent shown in the map. These indexes allow automation of map compilation for atlases and map series. An example of a location-specific parameter would be the projected coordinate system of a map on a given page of an atlas versus another page. An example of a sheet specific parameter would be the title or page number of given map sheet. There are many other kinds of parameters that can be used in location specific ways, like drawing instructions for layers. A useful example would be to display a combination of terrain and landcover as the background for a map series. Ideally the range of tints for the whole series would be determined and then only the applicable segment of those tints would be used for each map sheet's layer given the location shown in the map.
Tools	These are the functions and tools that process the GIS data into cartographic data.

Table 2

Relationship	**Description**
Colors and Symbols	The symbols in a given map dataset must relate to the colors in that map dataset such that the symbols only use the colors defined in that particular map dataset.
Elements and Symbols	The Elements listed for a specific map dataset must use the symbols that are defined for use in that same map dataset.
Data symbology and Symbols	The layers listed for a specific map dataset must use the symbols that are defined for use in that same map dataset.
Data Frames and Page Layout	The page layouts in a map dataset may only contain data frames defined for use in the same map dataset.
Cartographic Feature Classes to Tools	A Cartographic feature class needs to know which tool it was produced with. The inputs and parameters used by the tool will continue to be relevant information for the cartographic feature class because of its relationship to its source GIS data. Cartographic feature classes and tools that produce them must know about and continue to co-exist in the same map data set. This is not a data model requirement, but rather a usage context requirement. The idea is to make it possible for a system designer or project manager to have only relevant tools exposed in a given map data set.
Cartographic Feature Classes to GIS Data	When GIS data is used as input to a tool that produces a cartographic feature class certain properties of that GIS data become relevant, based on the particular nature of the tool, to the cartographic feature class. In particular, if the GIS data is updated such that its features with respect to those relevant properties change, it becomes possible for the corresponding

Relationship	Description
	features in the cartographic feature class to out of date. Thus, a relationship between the features in a cartographic feature class and its source GIS features must be maintained. ArcMap, will have software tools that can query the state of this relationship and tell map makers that certain cartographic features are out of date and depending on the methodology used to create those features, the map maker may have options to automatically re-run the tool for the affected features or run the tool again and reproduce the cartographic feature class, or manually edit the cartographic feature. The main idea is to keep important information that tools can provide when cartographic feature classes are produced and use that information to make subsequent data and map processing operations as automated as possible.

One important note about Table 2 is that objects like colors can be defined to be used within multiple map datasets such that a specific color object knows that it can be used by the symbols contained in specifically listed map datasets.

The mapping software also independently manages a number of relationships between objects. These include the state of map elements like where they are located on the page, for instance ensuring the title is centered on the page or index map updates to show the extent of the map. The GIS data may also adhere to a data model that has relationships within the data as well, however, these relationships should be considered independently of this discussion.

An important concept, cartographic feature classes, was introduced in table 1. Table two followed up with some specific behaviors for cartographic feature classes. Cartographic feature classes are one of the key ways that multiple representations, potentially at different scales are produced and managed in a multi-purpose Geodatabase. An important aspect of cartographic feature classes that should not be overlooked is that while they provide unprecedented flexibility, the tools that produce and edit them must be robustly implemented with the database storage method and ArcMap editing functionality accounted for in a systematic way.

With regard to producing and editing cartographic feature classes, there is larger context, which is that many tools are combined in a process model to produce a map. Sometimes changes in the GIS data can produce changes that dictate reprocessing all parts of the model after a particular point. ArcMap will be able to support this level of process automation[2].

[2] The Geoprocessing Framework that is currently under development at ESRI is a framework that will allow organizations to document and manage complex data processing operations. There are two basic kinds of processes, tools and models, which are created and used this geoprocessing framework. Tools are specific operations; they are given inputs and operational parameters and they produce output data. Models are a logical set of tools that accomplish a larger job. The data processing required to turn GIS data into a specific map can be encapsulated in a set of tools, which in turn can be saved within a logical sequence as a model. Such a model for producing the data for a map is useful for producing the most current depiction of the data for that map. This same approach is also useful in a larger context for managing production of updated areas in a map series or atlas.

Conclusions

Modeling the complete set of cartographic information needed to produce high quality maps with little or no human intervention is a goal that will be achieved in the near future. Some organizations are already modeling geographic and cartographic information in a database, and have become a force, driving ESRI to achieve this goal. There are however, other impedances to overcome. Mapping software is still evolving with respect to the range of possible ways to represent data, for instance a cartographer's eye, mind and hand are still unparalleled in their speed and creativity to judge and draw the most appropriate cartographic representations. Cartography has existed as an art and science for many centuries and the computer has only been widely used for decades. With ideas like those presented in this paper, the capabilities and intelligence of digital cartography software systems should evolve rapidly and will, perhaps within a few more decades encompass most or all of what has and can be done by hand, and perhaps drive much of the innovation in the field of cartography.

References

Hoel, E., Menon, S., Morehouse, S., *Building a Robust Relational Implementation of Topology*, http://downloads.esri.com/support/whitepapers/ao_/BuildingRobust Topologies.pdf, 2003

Nicewarner, M., DataModel.Org , http://datamodel.org/, 2003

Zeiler, M., *Modeling Our World,* ESRI Press, 1999