Title: Rapid Mapping Tool: an ArcMap Extension

Author(s): Steve P. Linger, Paul M. Rich, Doug Walther, Marc S. Witkowski, Marcia A. Jones, and Hari S. Khalsa

Rapid Mapping Tool: an ArcMap Extension

Steve P. Linger, Paul M. Rich, Doug Walther, Marc S. Witkowski, Marcia A. Jones, and Hari S. Khalsa

June 20, 2002

GISLab, Environmental Dynamics and Spatial Analysis Group
Mail Stop D452, EES-10
Los Alamos National Laboratory
Los Alamos, NM 87545 USA
tel: 505.667.7147, email: spl@lanl.gov, http://www.gislab.lanl.gov

Abstract
Cartographic production laboratories produce large volumes of maps for diverse customers. Turnaround time and consistency are key concerns. The Rapid Mapping Tool is an ArcMap based tool that enables rapid creation of maps to meet customer needs. This tool was constructed using VB/VBA, ArcObjects, and ArcGIS templates. The core capability of ArcMap is extended for custom map production by storing specifications associated with a map or template in a companion XML document. These specifications include settings and preferences used to create custom maps. The tool was developed as a component of an enterprise GIS, which enables spatial data management and delivery using ArcSDE, ArcIMS, Oracle, and a web-based request tracking system.

Introduction
Geographic information system (GIS) technology has revolutionized the process of professional cartography. Professional cartography involves building publication-quality maps that incorporate multiple geospatial data layers and various key map elements, such as titles, legends, scale bars, and neatlines. GIS software developers have addressed most of these basic needs, but only recently have they addressed broader process and organizational needs, such as storage of map specifications and finished products in a database. We need to rethink the process of cartography within an enterprise GIS framework, which ensures documentation, archive, and access of maps and map components. Herein, we present the design and implementation of a prototype mapping tool that expands the current capabilities available within Environmental Systems Research Institute (ESRI) products.

The GISLab at Los Alamos National Laboratory (LANL) has published more than ten thousand maps over eleven years for a diverse customer base. Customers include internal LANL projects and operations, as well as local government, state environmental agencies, regional pueblo governments, national agencies and departments, consultants, and other national laboratories. The maps represent many different disciplines, including environmental monitoring (air, soil, and water), environmental remediation, geology, ecology, and facilities management. These maps are used to support scientific projects and basic research fundamental to the mission at LANL.
To meet the needs of these diverse customers and produce a high volume of maps, GISLab employs standard methods and tools within an enterprise GIS. Documented procedures, hardware, software, and data comprise the enterprise GIS. Software used by GISLab include Oracle relational database management system (Oracle), and various ESRI products: Spatial Database Engine (ArcSDE), Arc Internet Map Server (ArcIMS), and ArcGIS (Rich, et al. 2002, Keating, et al. 2002). Additionally, custom applications and tools are developed as needed to meet special needs for GISLab and its customers. One such custom tool is the Rapid Mapping Tool (RM Tool).

The RM Tool is an ArcMap extension with the following features:

- extends the functionality of ArcGIS to create custom, variable-size maps;
- stores custom properties for maps and templates in XML (eXtensible Markup Language) documents as distributable, intermediate archives;
- simplifies the interface to map publication elements for novice users and rapid map creation;
- facilitates the transition from AML-based cartography to ArcMap-based cartography;
- uses Oracle as a permanent repository for map properties.

Like many professional GIS organizations, GISLab is faced with a substantial legacy commitment to Arc Macro Language (AML)-based cartography, but is now making the transition to cartography in the ArcGIS environment. The RM Tool is intended to replace capabilities currently provided by AML scripts. The tool serves as a prototype for porting the extensive capabilities of the existing, AML-based, map-making system to ArcGIS. The RM Tool requires input of specifications and data from a spatial data warehouse to produce professional quality maps (Figure 1).

Figure 1. The Rapid Mapping Tool (RM Tool) provides a user-friendly means to produce professional quality cartography.
**Design Requirements**

The RM Tool is designed for both experienced and novice GIS users. The tool presents a graphical user interface (GUI) that is simpler than the ArcMap menu system and provides shortcuts to the more commonly used map properties. For the novice mapmaker, the interface must be simple and easy to navigate. For the more advanced user, the interface must allow creating variable-size, or “VarSize,” maps, expedite map publication, and enable storing map properties in the Oracle repository.

The capability to create VarSize maps was first developed using AML tools ten years ago by GISLab staff and requires a non-standard approach for creation. The approach involves fixing not only the map extent but also the map scale for the primary data frame. This changes the physical size and location of the page, data frame, and all map elements. VarSize functionality must reposition each map element after resizing according to each element’s desired behavior.

Details for each published map, such as the title, date, extent, and cartographer, must be cataloged, not only for archival purposes, but also to enable query and retrieval. Questions such as “What data sources were used to make map number 131404?” and “What maps encompass a specific geographic coordinate and were published between Jan. 1, 2001 and Jan. 1, 2002?” need to be answerable. Oracle, the repository for GISLab enterprise data, stores the map information. By integrating the RM Tool with the spatial data warehouse and the request tracking system, information about each map produced can be made accessible through role-based interfaces to cartographers, customers, administrators, and project managers.

**Development Environment**

The advent of ArcGIS, and the extensibility that ArcGIS offers through the use of Visual Basic for Applications (VBA), Visual Basic (VB), and other COM (Component Object Model) compliant development languages, enables the creation of customized application functions with robust behavior. Developers can further exploit this development environment through the use of third-party products and free, downloadable code and COM components. The RM Tool uses the Microsoft (MS) XML Parser Version 3.0, a COM plug-in that can be downloaded at no cost from the Microsoft website. This component can be added to a VBA or VB project to embed development objects for creating and reading XML documents.

As an ArcMap extension, the RM Tool is a binary Dynamic Link Library (DLL). This ensures that scripts remain untouched by users, simplifies versioning, and allows functionality to be neatly contained within a single package. Though the final version of the tool is a DLL created using VB, initial prototyping and development employed VBA within ArcMap. Further into the development phase, the tool was ported to VB as an ArcMap extension.

**RM Tool Implementation**

The RM Tool interacts with software and data components such as ArcMap, Templates, XML, and Oracle (Figure 2).
Figure 2. The RM Tool was built within the ArcObjects framework and interfaces with the ArcMap and Template development objects. The tool uses XML to store properties and map specifications, and archives map data in an Oracle database.

Use of Map Templates

Map layouts commonly have many elements in common, such that a limited number of general layouts accommodate most needs. An ArcGIS Template can be created for each general layout. The cartographer can then choose an existing template containing the general desired layout as a starting point for creating a new map. The templates retain properties for a layout, such as whether grids and graticules have been created for a data frame, the location, size, color, and font for text elements, and similar properties for north arrows, geometric shapes, and scale bars. When invoked, the RM Tool retrieves and presents the primary template properties to the user through a tabbed dialog menu. The RM Tool automatically assigns numeric identifiers to each map element and enables users to give descriptive names to these map elements. The use of templates is optional with the tool; if no custom template is selected, the RM Tool interacts with the same ArcObjects interfaces through a default layout template.

Use of XML to Store Map Specifications

The RM Tool uses XML documents to archive user-selected behaviors and preferences that cannot be stored within an ArcGIS template using ArcObjects. Each XML document is coupled with an ArcMap document or template. The XML document type is used as a container for this information for two reasons. First, XML, as an industry standard, is interpretable by most programming languages. For VB, the Microsoft XML Parser component provides classes that can be used and modified to understand a specific set of XML tags. Second, XML documents can evolve to support new properties, objects, and settings without overhauling the XML document or the codes that interpret the XML format. This enables the XML document to expand and accommodate evolving needs of the tool (Figure 3).
Figure 3. XML is used to store map specifications. In this example, XML represents offset information for a map scale bar element. Map element locations are stored relative to the main map data frame to enable production of variable-size (VarSize) maps.

VarSize Maps

GISSLab publishes standard-size maps (e.g., 8.5”x11”, 11”x17”, and 24”x36”), as well as custom-size maps. VarSize maps represent approximately 75% of the maps produced by GISSLab. These maps differ from custom-size maps in that the physical map size for VarSize maps depends on the map extent and scale of the main data frame. Specifically, the cartographer sets the map extent for the primary data frame, and with the map extent held constant, adjusts the scale for the data frame. This causes the physical size of the data frame to change.

The overall page size of the map layout must change to accommodate the new size of the data frame, and the locations for many of the map elements must be adjusted in response to the altered page size. The map elements include legends, text elements, neatlines, additional data frames, scale bars, and geometric shapes. The desired behavior of each map element is defined by the user and archived in the XML document that accompanies the ArcMap or template document. Figure 3 shows sample XML code used to specify how the scale bar is to be located when producing a VarSize map. The scale bar is identified by a unique integer identifier, or “ID.” The ID is stored within the template or ArcMap document in the CustomProperty member of the IElementProperties interface. The map element ID is the link between the map element object in ArcObjects and the behavioral settings for that element, which are archived and managed using the XML document.

Figure 3 displays sample XML code and the corresponding graphical representation to describe how the scale bar should be positioned after the main data frame and map page have been resized to create a VarSize map. The scale bar (with ID 11) will be relocated so that its lower left corner is offset in the X- and Y-axis directions by the distances specified in the XML document from the lower right-hand corner of the main data frame. The circles highlight the lower right corner of the main data frame and the lower left corner of the scale bar.
Similarly, the VarSize properties for all map elements in a map layout can be archived in the XML document. In some cases, map elements are not relocated relative to the main data frame. The main data frame itself is one example. Also, neatlines are often associated with the entire map page instead of a data frame.

Storage of Map Data in a Database

To complete the cycle of map production, properties for each published map are archived in an Oracle enterprise database as a permanent record. Archived map properties include title, cartographer, date, data layers, map extent, requestor, map scale, time to complete, page size, and comments. Users can access and query these map properties through several GISLab application interfaces, including the request tracking system, the RM Tool, and an MS Access-based query and update tool that also connects to the Oracle repository (Figure 4).

![Diagram](image)

Figure 4. Role-based application interfaces provide access to map information archived in the enterprise database. Different users, such as customers, administrators, managers, and GISLab users, have access to different subsets of information, and appropriate privileges for query and data modification.

Enterprise GIS

Enterprise GIS integrates the flow of geospatial data with processes to enable geospatial data sharing by a large number of users (Figure 5). Through desktop applications and web-based interfaces, users can access map data and track the progress of specific projects through an enterprise request tracking system. Role-based data access is accomplished through commercial and custom applications to enable capabilities such as map production, project tracking, reporting statistics on maps produced, and assessing project costs for customers.
Conclusion

The RM Tool was developed for professional cartography in an enterprise GIS environment. It simplifies map production, adds functionality to ArcMap through custom map-making capabilities (such as VarSize maps), and provides a quick learning curve for novice users and experienced users new to ArcGIS. Implementation of this tool takes advantage of the use of ready-made COM components and the interoperability of ESRI ArcGIS products. The RM Tool was designed as an integrated component of an enterprise GIS, with essential map information stored in a database. As such, one can track the progress of each map produced from start to completion. In addition, one can readily query, access, and modify finished maps. This leads to the concept of an enhanced spatial data warehouse, where not only map data are stored and accessed, but also map specifications, scripts, map elements, maps themselves, and map metadata. The development of integrated mapping tools, for which the RM Tool serves as a prototype, are key for the future of map data warehousing.

Acknowledgements

Support for development of the RM Tool was provided by the Cerro Grande Rehabilitation Project (Task X046) and the Los Alamos National Laboratory Earth and Environmental Science Division. We thank Gordon Keating, Rick Kelley, Randy Mynard, and Thomas Riggs for valuable input and edits.

References


Walther, D., 2002. GISLab procedure for desktop cartographic production. GISLab Procedure GP-4.0, Los Alamos National Laboratory, Los Alamos, NM.