James Frew, Larry Carver, Christoph Fischer, Michael Goodchild, Mary Larsgaard, Terence Smith, and Qi Zheng

The Alexandria Rapid Prototype: building a digital library for spatial information

A first step toward the Alexandria Project's goal of a distributed digital library for spatially-indexed information is a "rapid prototype" system. The system was built in a period of six months, with a small team, almost entirely from commercial "off-the-shelf" software and hardware, at a single site, and populated with a limited set of heterogeneous spatial data. We are currently using the system to evaluate such issues as: implementing spatial metadata standards in a relational schema; using high-level scripting languages to customize large software packages and "glue" them together; and the relative merits of map-based versus forms-based user interfaces. These evaluations are helping drive the ongoing design of Alexandria's final deliverable, a distributed "testbed" system. The rapid prototype system is also serving as a staging area for data and metadata that will eventually be migrated to the testbed.

INTRODUCTION

The Alexandria Digital Library Project is a consortium of universities, public institutions, and private corporations headed by the University of California at Santa Barbara. Alexandria is one of six such digital projects jointly sponsored under a Digital Libraries Initiative by the National Science Foundation, the National Aeronautics and Space Administration, and the Department of Defense. The goal of the Alexandria Project (Smith 1994) is to build and deploy a distributed, online library for digital, spatially-indexed information, including maps, satellite imagery, digitized photography, and associated metadata. Currently, collections of such data are often non-digital (e.g., paper maps, photographs, atlases, gazetteers), and collections of a usable size and diversity are found only in the largest research libraries. These factors combine to make most spatial data a scarce and inaccessible intellectual resource. Hence a major goal of the Project is to make such resources available to a much broader community in the form of a distributed, digital library.

The main goal of this paper is to describe the architecture, implementation, and functioning of a rapid prototype system for the Alexandria Library. The experience gained in the construction of the Alexandria rapid prototype (ARP) is being used in the design and implementation of a main testbed system that is currently under construction. The paper is structured as follows. We first describe the goals and constraints that guided the design and implementation of the ARP. We then describe the functionality of the system from a user's perspective and the architecture of the system, which presents the developers' perspective. We end with a brief discussion of the next steps in the development of the Alexandria Digital Library. We note that in a companion paper (Fischer et al. 1995), we describe in detail the important issue of the design and implementation of a metadata schema for the ARP.

THE GOALS OF THE ALEXANDRIA RAPID PROTOTYPE

The ARP system has three major goals. First, and most important, it is the Project's tangible reference point for the current state-of-the-art in operational digital spatial data management -- i.e., the technology available today for assembling a "turnkey" system. As such, the ARP's software architecture is centered on a commercial geographic information system (GIS), and its hardware architecture is based on locally-networked workstations with modest (150 GB) shared magnetic and optical disk storage. The ARP's functionality should serve as a baseline against which the testbed can be measured.

The second goal of the ARP is to help us refine the specifications for several testbed components. For example, feedback from the ARP's mix of map-based and form-based queries (described below) has influenced the development of the testbed's user interface. Feedback from the ARP's metadata schema will influence the design of the testbed's catalog component. And the experience of loading actual (as opposed to simulated) data into the ARP has given us a much better idea of the relative difficulty of supporting a variety of data in the testbed.

The final goal of the ARP is to "jump-start" the process of creating Alexandria's online holdings. This ingest process is far more complicated than simply copying digital data from offline to online media. For each type of data to be supported (e.g., satellite imagery, digital line graphs, scanned aerial photographs), metadata must be extracted and entered, reformatting procedures must written and debugged, etc. The testbed system will not mitigate these activities, so there is no reason not to begin loading data immediately, migrating it from the ARP as the testbed becomes available.

Constraints on the Alexandria Rapid Prototype

The ARP was designed and implemented in six months by a relatively small team of individuals. These severe time and resource constraints are reflected in at least three fundamental development decisions that were made at the outset. First, the ARP software consists entirely of either commercial "off-the-shelf" products, or scripts written in high-level interpreted languages -- we wrote no low-level (e.g. C) code specifically for the ARP.

Second, we took advantage of substantial existing hardware and data resources to assemble and load the ARP. The UCSB Map and Imagery Library (MIL), which houses the ARP system, already possessed both an acceptable computing environment, and data holdings sufficient to load the ARP with nontrivial test cases.

Third, we limited the ARP system to a single installation. While it is possible to access the system remotely as an X11 client, neither the current host system nor the MIL's current network can support the severe computational and display demands of multiple simultaneous ARP users. Additionally, having only one ARP site made it much easier to manage the development effort.

USING THE ALEXANDRIA RAPID PROTOTYPE

Based on both empirical and theoretical considerations, it is possible to distinguish two major classes of queries for spatially-indexed information. From an empirical point of view, based on MIL's considerable experience in serving hardcopy spatial data to large (over 30,000 users annually) and diverse (university, governmental, industrial, general public) user population, the a following two classes of queries dominate the set of user requests with respect to spatially-indexed information: (1) users want to discover phenomena that occur at specified locations (we call this a "what's here?" query); and (2) users want to discover the locations at which specified phenomena occur (we call this a "where's this?" query). From a more theoretical point of view, if one views entities in the real world as possessing an essential "spatial projection" which is captured to some degree in spatially-indexed representations, one may query the non-spatial properties associated with a given spatial projection (what's here) or one may query the spatial projections associated with a given set of non-spatial properties (where's this) (see Smith et al. 1987; Smith et al. 1995).

Users may specify or understand geographic locations both directly ("latitude 34 degrees 25 minutes north, longitude 120 degrees 54 minutes west") and symbolically ("Campbell Hall, UCSB"). When dealing with a mix of such representations, a useful intermediate abstraction is the location's footprint, i.e., its boundaries drawn on a map. Thus both of the above representations can be displayed as points (or polygons, at higher resolutions) on the same map. This drives the requirement that, for any data granule (discrete object) stored in the ARP, the corresponding metadata must include the data's footprint.

Of course, merely noting which data exist and where they may be found is usually not sufficient; users want to examine the actual maps, airphotos, etc. Thus both queries imply a retrieval of data by the user, from wherever it is stored, into an environment in which the data may be browsed and analyzed.

The remainder of this section describes the user interfaces that the ARP presents for these activities.

Where's This?

The basic query interface provided by the ARP is a form into which the user types metadata values describing the desired data attributes (media type, means of acquisition, date acquired, etc.). This is the only interface provided for "where's this?" queries, since the phenomenological attributes being entered by the user may not have an unambiguous or easily-manipulated graphical representation. The query form does provide significant error checking; in many cases, the possible values for a particular field are few enough that they can simply be selected from a menu.

A "where's this?" query returns all the metadata for any data in the ARP whose metadata match the (range of) values specified in the query form.

What's Here?

For "what's here?" queries, we provide an additional, graphical interface comprising a hierarchy of base maps. The base maps are displayed starting at the lowest resolution (the entire Earth); the user then draws a rectangular region of interest with the workstation's pointing device. The selected region is then enlarged to fill the query window. This sequence may be repeated; as the resolution increases, additional details (e.g., state and county boundaries, rivers, etc.) are added to the current base map to keep the user oriented. Once the user is satisfied with the region of interest, its coordinates are copied to the query form, where they may be manually edited if necessary.

A "what's here?" query returns all the metadata for any data in the ARP whose footprints overlap the region of interest. We should note that a single query may (and most usually do) include both "where's this?" and "what's here?" components, since all graphical input is ultimately reflected to, and the query issued from, the form interface.

Evaluating Query Results

If any granules are located whose metadata satisfy a query, then the metadata are displayed as a table, or as footprints drawn on the base map, or both, according to the user's preference. Individual footprints are linked to individual rows in the table, so that selecting/deselecting one may highlight/hide the other. This is useful both for verifying a granule's location, and for disambiguating overlapping footprints.

In addition to (or instead of) the granule table, the user may request that a data type table be displayed. This table has a single row for each data type retrieved. For example, if a query finds 10 Landsat Thematic Mapper images, then the granule table contains a row for each image, whereas the type table contains a single row for all 10 images. This helps the user to both un-clutter the display, and to refine a subsequent query if the previous one was too broad.

Retrieving Data

At some point the user will presumably want to examine specific data granules. These can be retrieved simply by selecting the appropriate footprints, or rows in the granule table.

The default retrieval action is to display a browse image of the granule. Browse images are optimized for rapid retrieval, rapid display, and visual information content; this optimization is generally achieved by sacrificing spatial (and, for imagery, radiometric) information content believed to have minimal visual impact. A typical browse image is a small TIFF (Aldus Developers Desk 1992) or JPEG (JPEG 1993) file.

If requested after viewing the browse image, the system retrieves the original granule (if no browse image exists, as for example with tabular data, then this is the default retrieval action). The granule is displayed if the metadata indicate this is possible. The user is then prompted for the granule's disposition (save to a local file, send to a printer or tape drive, ship over a network connection to a remote system, etc.).

INSIDE THE ALEXANDRIA RAPID PROTOTYPE

The previous section presented a user's or external view of the ARP. This section describes the ARP from a developer's or internal perspective, first in terms of its functional architecture and then in terms of the components that support this functional architecture which include metadata, data, software, and hardware.

Functional Architecture

The functional architecture of the ARP is intended to reflect the main functionality required in supporting basic library operations. In particular, this architecture is a special case of the functional architecture of the main testbed system (Smith et al. 1994).

The main components of the architecture, apart the networking aspects, are a catalog component; a storage component; interfaces for both librarians and users; and an ingest component. The catalog may be viewed as both a database of metadatata and a search engine that is analogous to the catalog system in a standard library. The storage component, which corresponds to the stacks in a regular library, holds the main collection of library items, although reduced versions of some large items are stored in the catalog. Further details concerning the contents of these components and their implementation in ARP are described below.

The ingest component is intended to provide the librarian with mechanisms for adding new items to the store, extracting relevant metadata from the items, and cataloging them. The current ingest component of the ARP is primarily focussed on creating library items from existing digital files, which are then placed in the storage component. Some of these items are also used as background maps to support search at the user interface and to serve as sample data sources described by meta records. We are also employing a flatbed scanner for the injest of aerial photographs, maps, and limited text files. As described in more detail below, metadata meeting FGDC and USMARC standards is created for each these files, but we are also importing metadata files from an existing NASA database of imagery.

The import of metadata that already exists in digitial form requires a conversion of source fields into target fields. Datatype conversion and units of measurement conversion is typically involved in such a conversion process. If the minimal requirements for a metadata record are not fullfilled, additional metadata must be added manually.

Metadata: The Catalog

Since the metadata strategy and schema employed by the ARP is described at length in a companion paper (Fischer 1995), only a brief description is provided in this paper.

Metadata in the ARP are stored in a relational database management system (DBMS), in a schema we developed based on a hybrid of the Federal Geographic Data Committee metadata standard (FGDC 1994) and the USMARC standard for electronic library catalog interchange (Library of Congress 1976). Our schema uses FGDC as a baseline, and supports selected USMARC fields (specifically, those for maps and computer files) as non-overlapping extensions (i.e., USMARC fields were added to the schema only if they had no FGDC equivalents).

We have completed a entity-relationship model of the hybrid schema, which supports well over 1000 attributes. We currently use only about 80 of the attributes in the ARP implementation of the schema; however, the completed model allows us to trivially implement additional attributes as they are needed to support our ongoing data ingest efforts.

One of the most important attributes supported by the metadata is the bounding rectangle (in latitude-longitude coordinates) of the corresponding data granule. These rectangles support the footprint metaphor used by the map-based user interface.

Data: The Holdings

Our goals in selecting datasets to be loaded into the ARP were to achieve both breadth, in terms of the number of data types supported, and depth, in terms of the availability of multiple data types, and multiple instances of a single type, for a particular location.

To focus the depth requirement, we selected a subset of California (specifically, Santa Barbara, Ventura, and Los Angeles Counties), for which the MIL has an especially rich set of holdings. All data types that we loaded into the ARP included at least one instance overlapping this region. In addition to thematic depth, we were able to achieve some striking multitemporal layerings (for example, coincident

1920s-vintage aerial photography and 1980s-vintage satellite imagery).

We carefully selected datasets from a variety of data types to satisfy our breadth requirement. Raster data loaded into the ARP includes Landsat and SPOT satellite imagery, and scanned aerial photography. Vector data include the Digital Chart of the World (DMA 1992), the TIGER cultural features (roads, cities, etc.) coverages from the U.S. Bureau of the Census (U.S. Bureau of the Census 1992), and hydrographic data from the U.S. Geological Survey (U.S. Geological Survey 1993).

Software Components

The ARP was built primarily from three large software packages: the Sybase relational DBMS; the Tcl/Tk scripting language and user interface toolkit; and the ArcView GIS. Sybase is a well-known relational DBMS. We use a Sybase server, running on a remote host, to store and manage the ARP's metadata, using our previously-described hybrid spatial metadata schema. The modeling tool in which we developed the schema automatically generated the Structured Query Language (SQL) statements necessary to instantiate the schema in Sybase.

Tcl/Tk (Ousterhout 1994) is a "freeware" package with two components. The "Tcl" component is a high-level interpreted language, which we use as the master control language for the ARP -- everything is ultimately invoked from a Tcl script. The "Tk" component is a set of extensions to Tcl that correspond directly to X11 "widgets" (user interface elements). We used Tk to write the "where's this?" query form and miscellaneous other user interface components.

ArcView (ESRI 1994) is a GIS for PCs, Macs, and workstations. For our purposes, its principal advantage is that it incorporates a scripting language ("Avenue") that allowed us to modify its user interface. We used ArcView to implement the "what's here?" query, and for all non-text data display (base maps, data footprints, retrieved maps and images). We also take advantage of ArcView's ability to query the Sybase server.

Hardware Components

The ARP runs almost entirely on a single Digital Equipment Corporation (DEC) Alpha workstation in the MIL. In addition to a color X11 display, the ARP workstation hosts a 50 GB disk array and a 100 GB optical disk jukebox, so all ARP data are stored locally. Additional workstations and X terminals are available on the MIL's local Ethernet and have been used for software development and metadata entry, but are not required to run the ARP. Also on the Ethernet is a PC system with a high-resolution flatbed scanner; this system is the primary means by which we ingest aerial photographs into the ARP.

The only remote component of the ARP is the Sybase metadata server, which runs on an IBM RS/6000 workstation operated by the UCSB Computer Center, and is accessed over the UCSB campus Ethernet. Use of this remote server was strictly an economy measure, to avoid having to purchase, install, and maintain a DBMS for the exclusive use of the ARP.

THE NEXT STEPS

With the completion of the ARP, the Alexandria Project is now focusing its full attention on developing a distributed capability over the World Wide Web, as the next step in the design and construction of the main testbed system. While it is too early to "close the books" on the ARP, some clear lessons have emerged that will influence the testbed.

First, current GIS architectures are too monolithic and inflexible to build the kind of system we require. The ArcView software performed more than adequately within its own problem domain, but was unable on its own to support a full library interface. Current industry trends toward the unbundling and modularization of GIS software give us some hope here.

Second, large software packages that contain built-in extension languages are much easier to deal with than those that do not. The programmability of ArcView, via its Avenue language, was critical to the success of the ARP.

Third, high-level extensible scripting languages like Tcl/Tk are excellent tools for building integrated systems from otherwise isolated software packages. Coding at this "module coordination" level is far more productive than having to implement individual modules.

Finally, metadata management is the Achilles' heel of a digital spatial data library. The rich history of structured metadata management for traditional library holdings does not exist for spatial data -- the ARP schema is one of the first of its kind. We were thus compelled to spend what seemed like an inordinate amount of time constructing a coherent metadata schema from the current array of standards. And once the model was constructed, we had to spend additional time building interfaces for ingesting and validating metadata, and inserting it into the DBMS.

As noted previously, we are not abandoning the ARP. Although no new functionality is planned, it is currently a valuable resource, both for the Alexandria Project as "ingest engine", and for the MIL as working system that, even in its present form, represents a quantum leap beyond the way most map libraries currently manage and present their holdings. However, we plan to add substantially to its functionality in the next stage of Alexandria development on the World Wide Web.

User Interface Evaluation

We have made just enough progress toward a user interface for digital spatial data to realize that we still have a long way to go. For example, it is by no means clear that the map-form dichotomy in the ARP interface is fundamental, as opposed to an artifact of our immature vocabulary for graphically manipulating non-spatial data. More generally, we need concrete feedback to answer the basic questions of user interface evaluation: "Does the user get the information as requested?" and "Does the user avoid information which is not needed?"

Our user evaluation plan involves real-time interactive logging of user activities, using hypermedia tools to simulate the look-and-feel of the ARP. Conventionally, interactive logging is analyzed by deterministic measures of performance such as counting keystrokes, recording time units between system dialog and user response, etc. These types of analyses are useful, but they neglect user cognition. Interactive user logs collected for this project will be analyzed using Protocol Analysis, which has been shown to provide a rich source of information to formalize understanding about semi-structured and intuitive knowledge (Ericsson and Simon 1993). Protocol analysis is a systematic methodology for treating semi-structured behavior and dialogs as quantitative data.

Data subsets will be embedded to experiment with query and response functions for limited portions of the spatial archive. Running "underneath" the simulated interface will be a set of interactive logging mechanisms recording keystrokes and mouse placement, documenting response times, etc. The logging mechanism will include a dialog function to converse with the user in semi-structured question-answer mode. The dialog function will be triggered throughout the evaluation session, either by the system or by the user. System triggers will initiate dialogs during specific tasks (during file retrieval for example, or on completion of a query formation). User inactivity over a threshold timeframe will also trigger a system dialog, to ask the user about confusion, or task success, for example. Users will be able to initiate dialog as well, and encouraged to keep a journal of their activities and impressions. These dialogs will be saved in a relational database for subsequent processing and analysis.

ACKNOWLEDGMENTS

The Alexandria Digital Library Team would like to thank: Michele Aurand, Barbara Buttenfield, Pintsang Chang, Richard Ellis, Meg Gardner, Greg Hajic, and Keith Park (The Alexandria Digital Library); Kelly Chan, Jack Dangermond, and Clem Henriksen (ESRI); Rob Crombie (DEC); Arlene Allen, Glenn Davis, and Sue Woodill (UCSB Computer Center); and Debbie Donahue (UCSB Institute for Computational Earth System Science) for their particular contributions to the success of the ARP effort.

REFERENCES

Aldus Developers Desk (1992) TIFF revision 6.0. Aldus Corp., Seattle, WA.

DMA (1992) Digital chart of the world, edition 1. Defense Mapping Agency, Fairfax, VA.

ESRI (1994) ArcView 2.0c software, Alpha/OSF1 version. Environmental Systems Research Institute, Redlands, CA.

FGDC (1994) Content standards for digital geospatial metadata. Federal Geographic Data Committee, U.S. Geological Survey, Reston, VA.

Fischer, C. et al. (1995) Metadata standards for digital libraries. The Alexandria Digital Library, University of California, Santa Barbara, CA.

JPEG (Joint Photographic Experts Group) (1993) Digital compression and coding of continuous-tone still images. ISO/IEC 10918-1,-2.

Library of Congress (1976) Maps: a MARC format. Library of Congress Information Systems Office, Washington, DC.

Ousterhout, J. (1994) Tcl and the Tk Toolkit. Addison-Wesley.

Smith, T.R., D.P. Peuquet, S. Menon, and P. Agarwal (1987) KBGIS-II: A Knowledge-based Geographical Information System. International Journal of Geographical Information Systems 1(2): 149-172.

Smith, T.R. et al. (1994) Towards a distributed digital library with comprehensive services for images and spatially-referenced information. The Alexandria Digital Library, University of California, Santa Barbara, CA (document available via http://alexandria.sdc.ucsb.edu/proposal.html).

Smith, T.R., Su, J., El Abbadi, A., Agrawal, D., Alonso, A. and A. Saran (1995) Computational modeling systems. Journal of Information Systems 19(4): 1-54.

U.S. Bureau of the Census (1992) TIGER/Line 1992, the coast-to-coast digital map data base. U.S. Bureau of the Census, Washington, DC.

U.S. Geological Survey (1993) 1:100,000-scale Digital Line Graph (DLG) data, hydrography and transportation; area 12, central Pacific states. Optional format. U.S. Geological Survey, Reston, VA.

Work supported by NSF grant IRI94-11330 and by major product contributions from the Digital Equipment Corporation and the Environmental Systems Research Institute, Inc.

James Frew
The Alexandria Digital Library
University of California
Santa Barbara, CA 93106
Email: frew@alexandria.sdc.ucsb.edu