

A New Technology for Interactive Online Mapping with Vector Markup and XML

Ilya Zaslavsky
San Diego Supercomputer
Center
University of California
San Diego
zaslavsk@sdsc.edu

As Internet cartography matures from static map images to interactive and animated maps, and embraces extensive GIS functionality, the limitations of presenting Web maps as image files become obvious. In this paper, a new technology for Internet cartography is demonstrated that uses direct vector rendering in a browser to create highly interactive virtual maps from distributed sources of geographic data. This technology is made possible by the advent of XML (eXtensible Markup Language) and XML applications for 2D vector rendering such as VML (Vector Markup Language) and SVG (Scalable Vector Graphics). AXIOMAP – Application of XML for Interactive Online Mapping – is a Web map publishing kit and a customizable virtual map interface that allows for the display and manipulation of multiple point, line and area layers, database query, choropleth mapping, hyperlinking, map labeling and annotation. To render maps in a Web browser (Internet Explorer 5, in the current version), AXIOMAP generates VML shapes “on the fly” from XML-encoded geographic data that can physically reside on different servers. A thin client-side solution, AXIOMAP provides for better interactivity than traditional map server-based approaches. The paper explains the functionality of AXIOMAP, the technology behind it, and presents several applications. A free version of the software can be downloaded from www.elzaresearch.com/landv/.

INTRODUCTION

Within mainstream online mapping technology, servers generate maps as pictures in one of the standard raster graphic formats supported by graphical web browsers. Interactivity is accomplished by delivering an updated map image in response to user requests. Examples of the Internet map server technology include multiple web mapping sites powered by ESRI’s ArcView IMS, MapObjects IMS, or ArcIMS [ESRI, 2000], MapInfo’s MapeXtreme and MapXsite [MapInfo, 2000].

This typical architecture has an important advantage of being able to accommodate almost any client graphic browser (since client requirements are minimal) and provide access to significant GIS functionality and large databases residing on a server. However, as noted in Andrienko and Andrienko (1999) and Stynes et. al. (1996), such server-side solutions place severe restrictions on map interactivity. Additional technical limitations include:

- (1) Changes, even minimal, to a currently displayed map require a “round-trip” to the map server, resulting in lower update speed and increased network load, especially with large images (a common solution therefore is to keep the image size small);
- (2) Since each “map picture” has to be generated by the map server, scalability in terms of number of simultaneous users becomes an important issue. Efficient scalability solutions exist but they tend to be expensive;

- (3) Cumbersome object identification on a raster image and loss of visual quality during zoom are common.

At least the first two technical limitations of this mainstream model have important social implications. Organizations and companies willing to provide efficient interactive web mapping should be able to afford powerful servers and a high bandwidth uplink connection. Combined with the high cost typical of the map server software itself, this leads to a large number of producers and users of geographic data being excluded from interactive map publishing. The mainstream technology reinforces a de-facto "*digital mapping divide*" where global, regional, and municipal powers still determine which data are collected and shared with the public. Neighborhoods and communities, non-profit agencies and small companies, as well as organizations that rely on common hosting services to publish their Web pages, still have limited means of making an impact on how geographic information is collected and presented, and hence on decision-making processes that may affect them directly. Interestingly, this situation in Internet cartography echoes the concerns expressed earlier by Harley [1988] and Wood [1992]. The technology described in this paper provides an affordable way for communities to transform static maps into highly interactive documents which have the potential of being subversive, dynamic, and more culturally and politically charged.

Various client-server and client-side technologies for online mapping provide an alternative to the "picture case" described above (following the terminology from the OpenGIS "Web Map Server Interface Specification" [Open GIS Consortium, 2000]). Client-side solutions typically are implemented in Java, ActiveX or with the help of various specialized browser plug-ins for handling vector graphics. Mapping software is either pre-installed on the client computer, or is downloaded each time along with map information. Large download size for software (1 MB on average, as noted by Kendall, 1999), the possible incompatibility with a client computer, and the potentially large size of data files to be transmitted over network are the major drawbacks of this solution. However, manipulating vector data on the client side allows for enhanced interactivity and performance advantages due to a reduced number of server requests. Excellent examples of this technology are given by the CIESIN's Demographic Viewer [CIESIN, 2000] and the Descartes system [Andrienko and Andrienko, 1999] developed as Java applets, GeoMedia Web Map [Intergraph, 2000] and AutoDesk MapGuide [Autodesk, 2000] which require special plug-ins. Compared to these examples, the technology described in this paper represents, in a sense, a "thin" client as it relies on vector rendering and XML capabilities of the browser and does not require any Java/ActiveX applications or plug-ins, either pre-installed or downloaded with the data.

The choice of a Web mapping architecture depends on many factors. These include bandwidth, price, graphic quality, interactivity, and the server and client side compatibility considerations mentioned earlier, as well as the needs of a particular online mapping application. If most user requests expect a new set of geographic features to be displayed each time on the client side (mapping the vicinity of a user-specified address, driving directions, etc.), the advantages of client-side caching of vector coordinates are minimal or non-existent. If user requests repeatedly ask for a map with the same or similar geographic features, re-using vector geometry information already cached on the client can lead to significant performance and interactivity gains (in atlas and thematic mapping

applications with a relatively large amount of attribute data. For such applications, a vector-based client-side solution appears more appropriate, especially since most graphic browsers provide automatic caching mechanisms that speed up access to recently used data.

With the advent of XML and XML-based vector graphics languages, a new and interesting mechanism for client-side vector mapping became possible. The Vector Markup Language as implemented in Internet Explorer 5 allows for encoding and scripting of inline graphic primitives in standard HTML files, and vector rendering in a browser without any special plug-ins or large software downloads. The project described in this paper implements this latest trend in the form of a light client-side interactive mapping system called AXIOMAP: Application of XML for Interactive Online Mapping [ELZA Research, 2000].

The following section reviews the emerging XML-based standards for representing and rendering geographic data, followed by a functional description of AXIOMAP in the second section. Applications of AXIOMAP, as well as its potential as a user front-end for a distributed mediated GIS are discussed in the last section.

EMERGING WEB STANDARDS FOR GEOGRAPHIC DATA

Recommended by W3C [W3C, 1998a], XML (eXtensible Markup Language) emerges as the new *Lingua Franca* for data interchange on the Web. It provides for semantic tagging of hierarchically-organized semi-structured data (i.e. data with changing structure, omissions and repetitions – which is often the case with geographic data). XML files are editable text files, and provide a flexible platform-independent format for exchanging information between heterogeneous systems over the Internet. Most database vendors already support or plan to support XML. In the realm of computer mapping and GIS, the interest towards XML is also growing, with major vendors such as ESRI, MapInfo, Intergraph, Oracle, CadCorp, LaserScan etc. starting to offer XML solutions as part of their Web GIS suites (company web sites and private communication).

Within the Open GIS Consortium, a focus on XML has been expressed, for example, in defining Web map server capabilities in XML [Open GIS Consortium, 2000], and representing the OGIS simple features [Open GIS Consortium, 1997] in the form of Geography Markup Language (GML). While the system described here was developed prior to release of GML drafts, it adheres to similar principles. AXIOMAP uses a simplified DTD (Document Type Declaration) to represent geographic feature collections as XML documents. Geographic features are exported into XML from a desktop GIS (ESRI's ArcView 3) and rendered in a browser as VML shapes. Compared to current GML DTD which is tuned to platform-independent storage and exchange of geographic data, the AXIOMAP DTD is tuned to efficient VML rendering of interactive maps (as discussed in the next section). However, recasting GML files in AXIOMAP DTD, and vice versa, appears straightforward with XSLT [W3C, 1999] as the DTD differences are not dramatic.

A very important recent development for Web cartography is a number of XML-based languages for 2D vector graphics proposed to the World Wide Web Consortium (W3C). Among them, the Vector Markup Language (VML), Precision Graphics Markup Language (PGML) and, most recently, the Scalable Vector Graphics (SVG) formats provide a mechanism for encoding and scripting graphic primitives for rendering in a browser. PGML (derived to a large extent from Postscript) and VML (with a simpler rendering engine) exist as W3C proposals, while SVG [W3C, 2000], a joint development of the two teams initially behind VML

and PGML, is close to becoming a W3C recommendation (at the time of writing).

VML [W3C, 1998b] is implemented in Microsoft's Internet Explorer 5 and is the graphics interchange format within the Microsoft Office 2000 suite. In contrast to VML, using SVG currently requires a special viewer. Several SVG viewers have been developed [IBM, 2000; CSIRO, 2000], and a browser SVG plug-in has been recently released by Adobe [Adobe, 2000]. With further adoption of vector graphic formats in browsers and the development of export mechanisms from popular GIS into these formats, VML and SVG have the potential of making a revolutionary change in the way geographic information is displayed and used on the Internet.

The AXIOMAP system (ELZA Research, 2000) consists of two modules. The first module is an ArcView extension for converting ArcView themes into XML files conforming to AXIOMAP DTD. The second module is a map viewer application written in Javascript, with the use of browser DOM (Document Object Model) and Cascading Style Sheets. The entire package can be downloaded from <http://www.elzaresearch.com/landv/>.

AXIOMAP: OVERVIEW OF THE TECHNOLOGY

To publish a particular interactive map on the Internet, one needs to create a set of XML files by running the ArcView extension, and copy them, along with the map viewer, to a Web server. The XML files contain geometric (vertex coordinates of features), attribute and rendering information for each map layer. In addition, the ArcView extension creates a project organization file with names, locations and rendering styles of all layers registered to the map, and general map rendering instructions (map title, extent, annotation, projection, coordinate translation scheme, several GUI elements for the initial map view, etc.). When creating the collection of XML files, a map publisher specifies:

- **Rendering** style or symbol for a layer. By default, rendering styles are carried over from ArcView projects. However, the styles can be easily changed in a standard text editor by editing the XML files that contain rendering information.
- A variable to be used for **labeling** features. In the map viewing application, the feature labels can be turned on or off, or permanently attached to any part of the map. Feature labels (for thematic maps – along with attribute information) also appear as the user positions the mouse pointer over a feature.
- An optional variable containing **URLs** of pages to be opened when a user clicks on a particular map feature. The URLs can point to local or remote documents, or represent database requests, e-mail instructions, etc.
- **Location** of XML files representing a theme. The files for each layer can physically reside on a local computer or on one or several servers. In a distributed computing environment this ability allows one to maintain and update cartographic layers individually, thus ensuring that information displayed on a map is as current as possible.
- **Grouping** of different variables for effective presentation in the interface. This feature allows map publishers to provide user access to a large number of variables and themes from a relatively compact interface (in the example available at <http://www.elzaresearch.com/landv/demo5/main.htm>, users have

- access to 300 thematic variables and 200 point layers describing physical conditions of buildings and streets in a neighborhood).
- **Color scheme** of the map viewer interface.
 - **Language** of the interface and system messages. Currently, AXIOMAP supports English, Spanish, French, Russian and Chinese languages (all language-specific information is stored in yet another XML file, so changing a language requires translation of only this file).
 - **Coordinate system.** Spatial data can be converted from any coordinate system in ArcView into XML. However, distance buffers around point features will be enabled only for themes in UTM or latitude/longitude coordinates. For these coordinate systems, cursor position will be displayed in meters or decimal degrees respectively.

The structure of XML files ready to be served as an interactive map is shown in Figure 1. Each map resides in a separate folder containing at least an XML file with project organization information, metadata description, and a standard frameset and program calls. XML files for individual layers can reside in the same directory, or anywhere on the Web, as referenced from the project file.

Each XML file representing a particular layer contains coordinate strings for geographic features in this layer. For example, XML files for area layers represent a list of area records each of which contains the area ID, name and strings of vertex coordinates for polygons that compose the area. For the base area layer, area records can be linked to one or more attribute datasets also possibly residing on different servers as XML files. A fragment of an XML document with geometric information about San Diego neighborhoods is shown in Figure 2.

For the sake of rendering efficiency and small file size, feature coordinates stored in XML files are encoded as integer values with one or two more significant digits than maximum pixel resolution of a display. For example, with typical screen resolutions up to 1600 pixels, coordinate encoding in four-six significant digits provides enough visual quality during zooming. The coordinate translation scheme contained in the

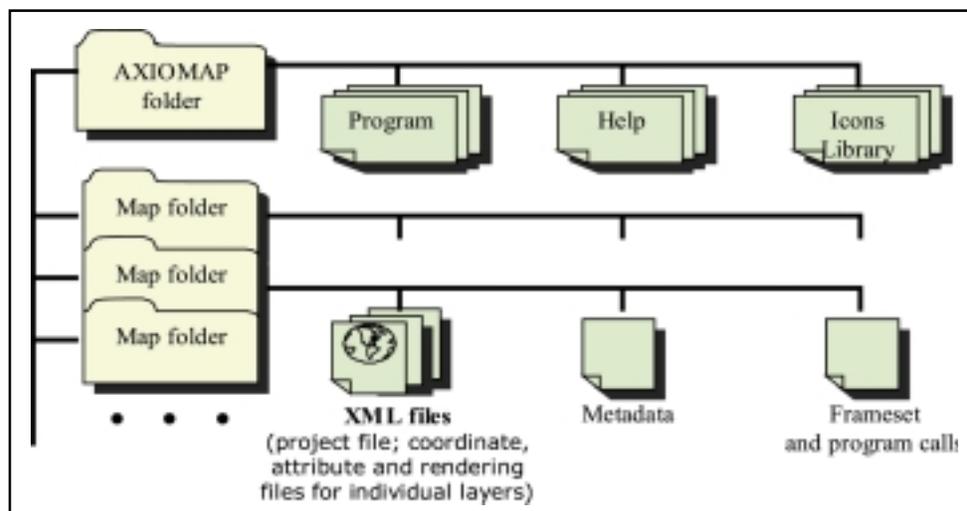


Figure 1. File organization of the AXIOMAP interactive map viewer.

```

<?xml version="1.0" encoding="iso-8859-1"?>
<rs>
<r><name>Horton Plaza</name><URL>horton.htm</URL><labelpos>41.46,77.51</labelpos>
<c>5076,1540 4986,1540 4895,1539 4803,1539 4715,1539 4622,1539 4534,1538 4534,1641
4534,1745 4534,1856 4622,1856 4711,1856 4800,1856 4893,1855 4984,1855 5075,1854 5075,1749
5076,1646 </c></r>
<r><name>Gaslamp</name><URL>gaslamp.htm</URL><labelpos>44.60,83.00</labelpos><c>5162
1013 5084,1057 5083,1116 5081,1222 5079,1326 5079,1433 5076,1540 5076,1646 5075,1749
5075,1854 5167,1854 5257,1855 5257,1750 5259,1647 5260,1541 5262,1434 5262,1328 5263,1222
5263,1013 </c></r>
...

```

Figure 2. A fragment of an XML document with information about two San Diego neighborhoods. Each neighborhood has a label (name), label position, an optional URL, and a list of coordinate pairs.

map project file allows the re-computation of these numbers into approximate geographic coordinates “on the fly”.

The second part of the AXIOMAP system is the viewer application which represents a DHTML client allowing users to construct multi-layer map presentations and control various aspects of map appearance. The viewer (its snapshot is shown in Figure 3), though less than 65K in size (less than 50K if the left frame of the viewer is pre-generated), allows users to:

- Perform instant choropleth mapping using datasets from different servers (controlling number of intervals and classification method). Users can also construct new variables as arithmetic functions of existing variables belonging to different datasets, and map them;
- Display/hide multiple point, line and polygon layers; label geographic features. These operations (as well as choropleth mapping) are accomplished without page refresh or map redraw, which enhances the user interaction experience. Particular map layers are fetched from Web servers only if they are requested by the user;
- Display, analyze and save attribute information for any object or a group of selected objects;
- Construct simple and incremental queries (based on multiple variables) and map query results;
- Navigate any map by zooming and panning, with user-defined zoom parameters. For better visual perception, different elements of map display scale in different proportion to the zoom level. For example, the size of point symbols changes logarithmically with the zoom level;
- Connect with Web sites, databases, or local pages associated with geographic objects;
- Analyze proximity of various objects to one another, by drawing circles of specified radii and color, anywhere on the map;
- Control the position and content of the legend and annotation information; and
- Print out map documents along with accompanying statistical tables.

Several applications utilizing VML for map rendering on the Internet have been developed already [Gaborit, 2000; Geotask AG, 2000;

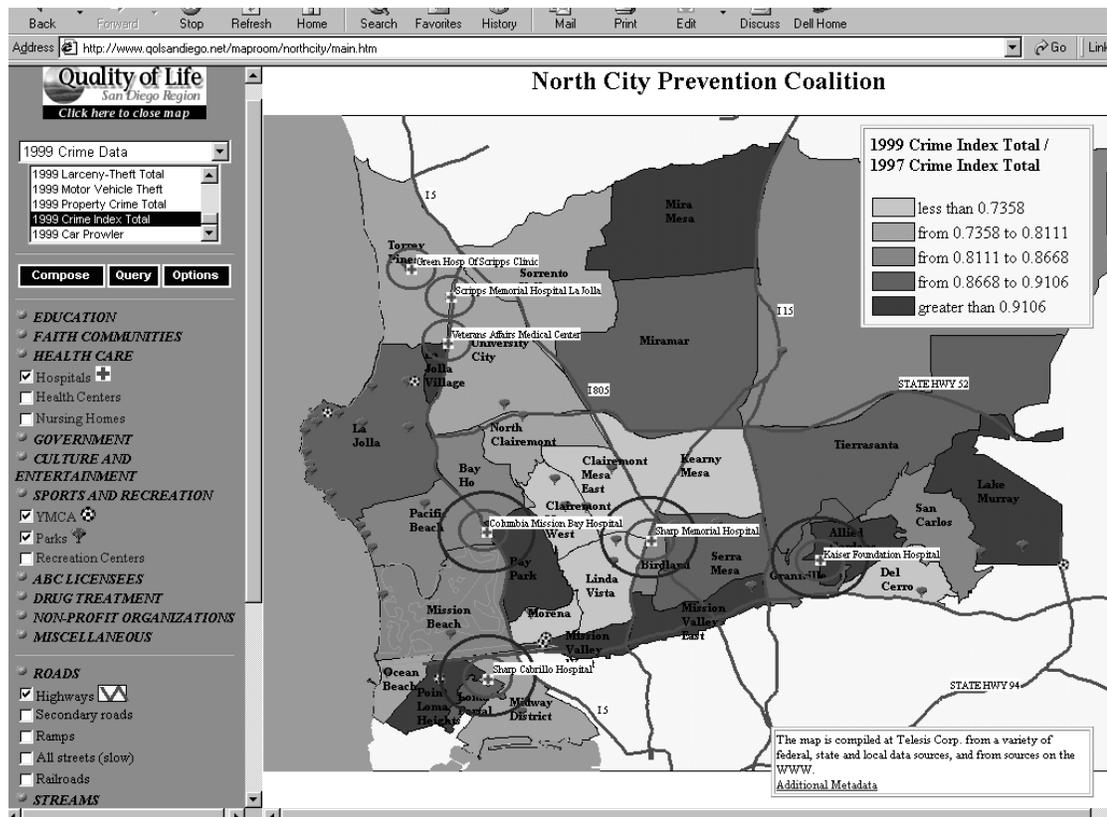


Figure 3. A snapshot of the AXIOMAP map viewer (data from the San Diego Quality of Life project).

Groundcontrol, 2000; XYZ Systemas, 2000]. A particular feature of AXIOMAP is that graphic elements, in VML format, are generated and displayed in a browser client “on the fly.” An example of a VML fragment generated from the XML fragment in Figure 2, is presented in Figure 4. A dynamic virtual map represents a “layered cake” of VML snippets which can be individually created and manipulated, rather than a physical VML document. This reflects the difference between two architectures for Web vector mapping, named the “graphic element case” and the “data case” (or the “feature case”) in the already cited OpenGIS document [Open GIS Consortium, 2000]. In the “graphic element case,” graphic primitives of a vector map are pre-built or generated on a map server and sent across the network to the rendering client. In the “data case,” server activity is limited to simply fetching data files on user requests, so that vector coordinates travel across the network and get converted to graphic primitives on the client side. The second approach, in our experience, provides for more latitude in defining interactive functionality of a web map, and results in smaller file sizes and easier attribute manipulations. With this architecture, implemented in the AXIOMAP viewer, meaningful and efficient user interaction with vector map data in the browser can be accomplished with much more modest computational resources compared to both “picture” and “graphic element” cases.

Since AXIOMAP works on the client side, it can be served from any Internet or Intranet server (in fact, it can even be used without a Web server). This flexibility comes at a price, however: for AXIOMAP to work the client browser must support XML/VML (i.e. it has to be Internet

```

<html xmlns:v="urn:schemas-microsoft-com:vml"><head><style>v\:* { behavior:
url(#default#VML); }</style>
...
<v:shape id="a1" class=z style="left:0;top:0;width:11590;height:7547" fillcolor="#FFFF99"
strokecolor="#000000" strokeweight="1" u="horton.htm" title="Horton Plaza" path="m5076,1540l
4986,1540 4895,1539 4803,1539 4715,1539 4622,1539 4534,1538 4534,1641 4534,1745 4534,1856
4622,1856 4711,1856 4800,1856 4893,1855 4984,1855 5075,1854 5075,1749
5076,1646xe"/><v:shape id="a2" class=z style="left:0;top:0;width:11590;height:7547"
fillcolor="#FFFF99" strokecolor="#000000" strokeweight="1" u="gaslamp.htm" title="Gaslamp"
path="m5162,1013l 5084,1057 5083,1116 5081,1222 5079,1326 5079,1433 5076,1540 5076,1646
5075,1749 5075,1854 5167,1854 5257,1855 5257,1750 5259,1647 5260,1541 5262,1434 5262,1328
5263,1222 5263,1013xe"/>
...
<div class=label style="top:75.51%; left:39.46%;">Horton Plaza</div><div class=label
style="top:81%; left:42.6%;">Gaslamp</div>

```

Figure 4. Fragments of the VML file generated by the AXIOMAP viewer from the XML document in Fig. 2.

Explorer 5), or – in the most recent cross-browser version – an Adobe SVG plug-in must be installed on the client side. While currently such client requirements limit the universal usability of this approach, standardization efforts on the side of W3C (especially as regards Web vector graphics, XML, DOM and ECMAScript) must improve the situation. At the same time, the AXIOMAP approach makes publishing interactive maps on the Web extremely simple, and provides for enhanced user interaction with map content.

AXIOMAP features and flexibility made it a useful instrument in a range of projects. The two examples below describe a research testbed for spatial information mediation being developed within the MIX (Mediation of Information using XML) project of the San Diego Supercomputer Center, and a community mapping portal developed as part of the Quality of Life in San Diego project. In the first example, AXIOMAP works as a user front-end on top of server software that generates XML files in the process of responding to user queries. In the second example, AXIOMAP is used as a stand-alone viewing and analytical interface to multiple point, line and area layers stored as XML files.

APPLICATIONS

AXIOMAP within MIX: interactive mapping of spatial query responses

The goal of the spatial mediator system is to give a user the ability to issue a single query to access multiple heterogeneous geographic sources, retrieve different pieces of the result, and seamlessly assemble these pieces into a composite response. Several earlier publications have focused on technical aspects of this wrapper-mediator framework for logical integration between geographic information sources [Gupta et al., 1999, 2000; Zaslavsky et al., 2000]. Figure 5 shows the general architecture of such a framework.

In this system, user requests are formulated from a web map interface (such as AXIOMAP) and sent to a *mediator*. The mediator generates a query evaluation plan, decomposes the request into small fragments and dispatches them to the appropriate sources of geographic data and services. The data and services, in turn, are “wrapped” into conversion software (a *wrapper*) that translates request fragments into the language of a particular source (a sequence of Avenue instructions in our experi-

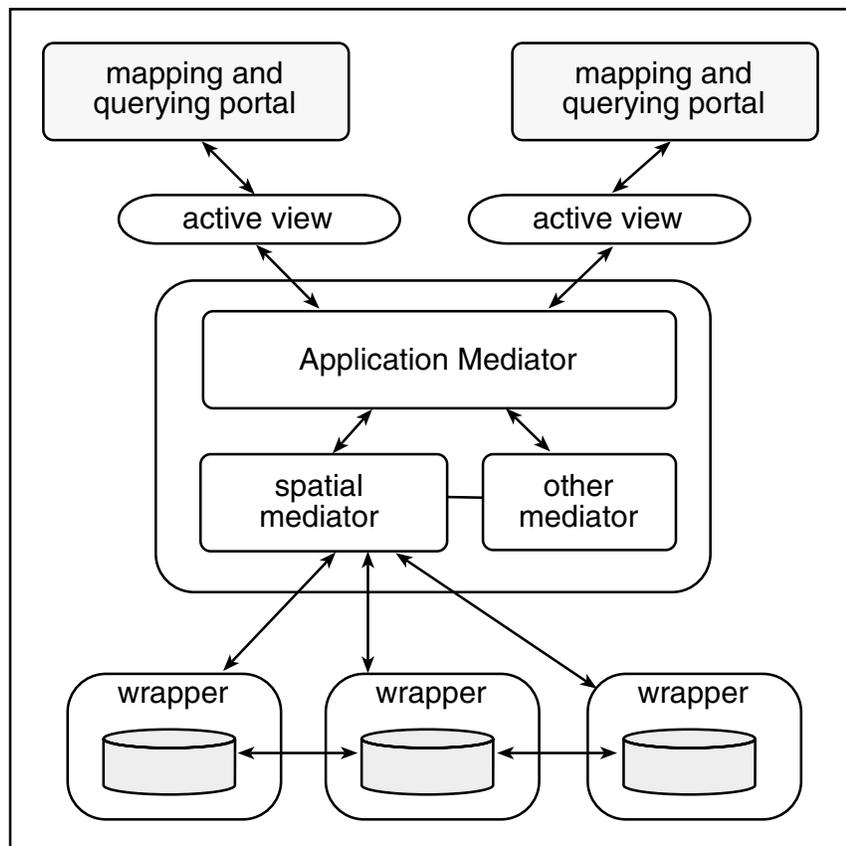


Figure 5. The architecture of a system of mediators and wrappers for heterogeneous information integration.

mental wrapping of ArcView sources, for example). The sources process the scripted requests and return the query results to the user as XML files. In our case, the source wrappers generate XML files from geographic layers requested in a query, within the spatial extent of the query result. At the same time, the mediator generates an AXIOMAP project file referencing the locations of these newly-generated XML layers. The AXIOMAP viewer then opens the project file and presents it to the user as a layered virtual map that can be further manipulated and analyzed interactively.

As an example, consider the following hypothetical spatial query:

Find neighborhoods in the North Coastal part of San Diego with median household income over 50K, without recreation facilities (recreation center, YMCA), and crossed by an interstate highway.

Generating an evaluation plan for a query like this, posed against a series of heterogeneous spatial data sources, is described in [Gupta et al., 2000]. In our example, suppose, source A contains a layer of San Diego neighborhoods with accompanying census information, source B has a digital map of interstates, and source C provides point layers for recreation facilities. Responding to the query, source wrappers generate XML files for each of the layers requested in the query (in our case, the neighborhoods layer is generated at source A as *a.xml* and *b.xml*, the interstate layer is *l0.xml*, and the point layers are expressed as *p0.xml* and *p1.xml*). In addition, one of the source wrappers (the last one in the query evalua-

tion plan), generates query results as *f0.xml* file. The mediator then generates a project file, a fragment of which is shown in Figure 6.

As a response to the query, the user can see the map interface shown in Figure 7. Initially, only the result of the query is displayed on the map, but the user has the option of turning on other layers referenced in the query. Additionally, the user can generate thematic maps, add buffers and labels, and use other AXIOMAP viewer capabilities to further customize the map.

```
<layer layertype="base"><source>http://www.sourceA.com/a</source>
<v><name>1990 Census Data</name>
<source> http://www.sourceA.com/b0</source></v></layer>
<layer layertype="line">>
<source> http://www.sourceB.com/l0</source><id>0</id>
<name>Interstates (Source B)</name>
<color>#ff0000</color><width>2</width></layer>
<layer layertype="point">
<source> http://www.sourceC.com/p0</source><id>0</id>
<name>YMCA (Source C) </name><icon>p0.gif</icon></layer>
<layer layertype="point">
<source>http://www.sourceC.com/p1</source><id>1</id>
<name>Recreation Centers (Source C)</name><icon>p1.gif</icon></layer>
<layer layertype="foregr">
<source>http://www.sourceA.com/f0</source></layer>
```

Figure 6. A fragment of a project XML file generated by the spatial mediator for the sample query.

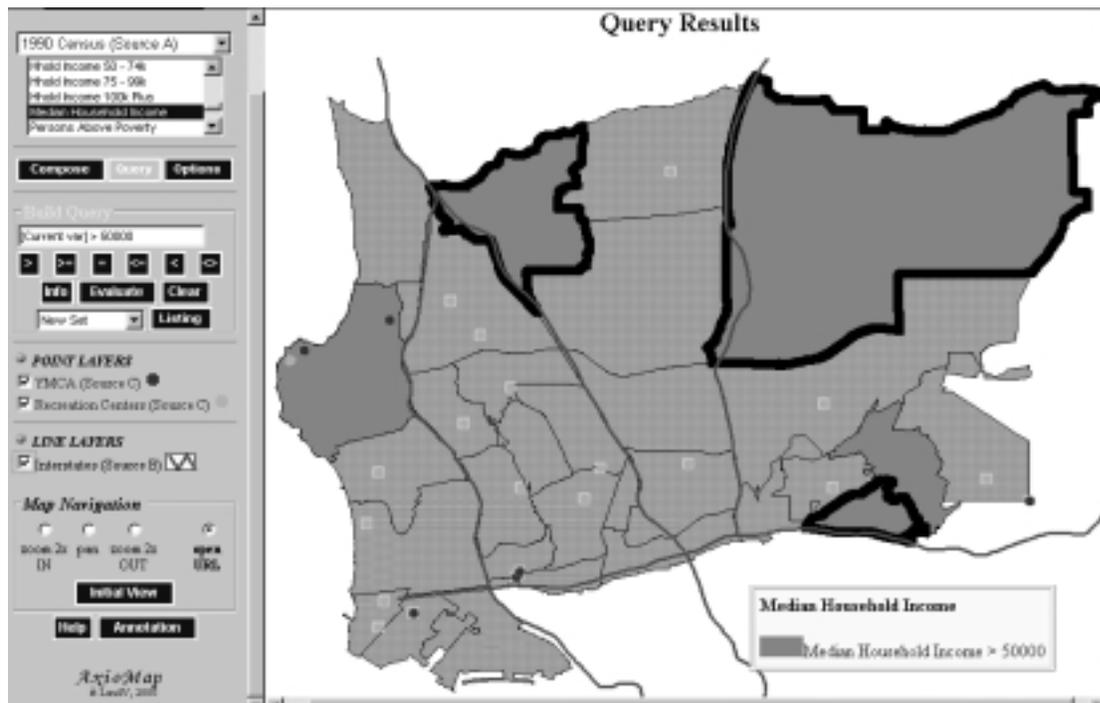


Figure 7. AXIOMAP user interface, with query results generated by a spatial mediator. The query results are shown as a black outline against other layers in the query.

AXIOMAP in the San Diego Quality of Life (QOL) Project

The San Diego Quality of Life Project is the core activity of a public partnership (including state and county agencies, foundations, universities, and non-profit organizations) with the mission to empower local communities through information sharing, development of QOL indicators and tracing their dynamics in each neighborhood, and statistical analysis and mapping of QOL indicators in a form accessible by community members. One of the main distribution channels for the QOL interactive maps and warehouse data is a special Internet community portal being developed by non-profit Telesis Corp. [Telesis, 2000]. From this portal, community users can access a wealth of information about their neighborhoods, including 11 interactive maps of San Diego Community Collaboratives (as in Figure 3), as well as more detailed community mapping projects. Each Quality of Life map enables community users to develop choropleth maps of about 300 variables (demographics, housing, ethnic and income composition, occupation and employment, land patterns, crime, forecast data from a local planning agency, etc.) and any arithmetic combinations of these variables, and display three dozen line and point layers (roads, hospitals and recreation centers, parks, drug treatment centers, government agencies and non-profits, churches and community centers, libraries and museums, etc.). The Quality of Life maps are accessible from the QOL web portal at www.qolsandiego.net.

For several key problem areas, a complete physical inventory of the neighborhoods was conducted, where about 200 types of point objects indicative of the quality of life conditions were recorded with a GPS-enabled mapping system. The recorded locations included areas with graffiti, drug paraphernalia, broken glass and trash, as well as physical conditions of each building (see <http://www.elzaresearch.com/landv/demo5/main.htm>). With these data, never before available in the form of an online interactive map, communities can support policies focused on local police protection and school security, home improvement funding and substance abuse prevention activities, etc.

CONCLUSION AND OUTLOOK

Compared to server-side mapping architectures, XML encoding and rendering of geographic information gives a different response to the trade-offs of online mapping, emphasizing dynamics, interactivity and interoperability for geographic data. At the same time, direct XML-based vector rendering in a browser avoids the often "heavy" Java/ActiveX/plug-in client-side solutions, thus remaining within the realm of what is typically called a "thin" client.

Since AXIOMAP manipulates vector coordinates and attributes of geographic features within a browser, this technology appears to be especially suitable for dynamic and animated user-driven map presentations, making use of dynamically extended graphic variables (*moving, rotation, oscillation, pulsing* – as in Holmberg, 1994). Additional research issues emerging from this experience include the development of virtual maps from distributed and independently maintained data sources, and analysis of novel online map publishing paradigms, as well as the social implications of online mapping.

REFERENCES

Adobe Corp. (2000) Scalable Vector Graphics (<http://www.adobe.com/svg/>)

- Andrienko, G. L., and N. V. Andrienko (1999) "Interactive Maps for Visual Data Exploration," *IJGIS* 13(4), June 1999, pp. 355-374.
- Autodesk, Inc. (2000) Autodesk MapGuide. (<http://www.autodesk.com>)
- AXIOMAP: Application of XML for Interactive Online Mapping (2000) (<http://www.elzaresearch.com/landv/>)
- Center for International Earth Science Information Network (2000). Demographic Data Viewer: (<http://www.ciesin.org>)
- CSIRO (2000) CSIRO SVG Toolkit (<http://sis.cmis.csiro.au/svg/index.html>)
- Environmental Systems Research Institute (2000). (<http://www.esri.com>)
- Gaborit, G. (2000) A VML demo (online at: <http://perso.wanadoo.fr/prosper/carto/demo.htm>)
- GeoTask AG (2000) GeoTask VML Server (<http://www.geotask.ch/images/VML/general.html>)
- Gifford, F. (1999) "Client vs Server -Which Side is Right for You?", *Mapping Awareness*, August 1999.
- GroundControl GeoTechnologies Inc. (2000) (<http://www.groundcontrol.ca/vmldemo/vmldemo.htm>)
- Gupta, A., Marciano, R., Zaslavsky, I., Baru, C. (1999) "Integrating GIS and Imagery through XML-Based Information Mediation". In P. Agouris and A. Stefanidis (Eds.) *Integrated Spatial Databases: Digital Images and GIS*, Lecture Notes in Computer Science, Vol. 1737.
- Gupta, A., Zaslavsky, I., and Marciano, R. (2000) "Generating Query Evaluation Plans within a Spatial Mediation Framework," *Proceedings of the 9th International Symposium on Spatial Data Handling*, Beijing, China, August 10-12, 2000.
- Harley, J.B. (1988) "Maps, knowledge and power." In D. Cosgrove and S. Daniels (Eds.) *Iconography of Landscape: Essays on the Symbolic Representation, Design and Use of Past Environments*, pp. 277-311. Cambridge: Cambridge University Press.
- Holmberg, S. (1994) "DML, a Map Language for Dynamics," *Proceedings of the 1994 European GIS Conference*, pp. 860-868.
- IBM (2000) IBM SVG Viewer (<http://www.alphaworks.ibm.com/tech/svgview>)
- Intergraph Corp. (2000) GeoMedia Web Map. (www.intergraph.com).
- Internet Mapper (2000) XYZ Sistemas Industriales. (www.imapper.com)
- Kendall, G. (1999) "A Guide to Internet Mapping Products and Pricing", *Mapping Awareness*, August 1999.

MapInfo Corp. (2000) MapXtreme, MapXsite (www.mapinfo.com).

Open GIS Consortium (1997) *OpenGIS Simple Feature Specification For OLE/COM* (Revision 1), Wayland, Massachusetts, 1997.

Open GIS Consortium (2000) *OpenGIS Web Map Server Interface Implementation Specification*. (<http://www.opengis.org>).

Peterson, M.P. (1995) *Interactive and Animated Cartography*. Englewood Cliffs, NJ: Prentice Hall.

Stynes, K., J. Wood, J. Dykes, P. Fisher, and D. Unwin (1996) "Publishing Cartography on the Web," Working Paper, International Cartographic Association Commission on Visualization. (online at: http://www.geog.psu.edu/ica/icavis/ICAVIS_working.html)

Telesis Corp. (1999) The San Diego Quality of Life Project. (<http://www.qolsandiego.net>).

W3C (1998a) *Extensible Markup Language (XML) 1.0*. W3C Recommendation, 10-February-1998. (<http://www.w3.org/TR/1998/REC-xml-19980210>).

W3C (1998b) *Vector Markup Language (VML)*. World Wide Web Consortium Note 13-May-1998. (<http://www.w3.org/TR/NOTE-VML>).

W3C (1999) *XSL Transformations (XSLT), Version 1.0*. W3C Recommendation, 16 November 1999 (<http://www.w3.org/TR/xslt>)

W3C (2000) *Scalable Vector Graphics (SVG) 1.0 Specification*. W3C Candidate Recommendation, 2 August 2000 (<http://www.w3.org/TR/2000/CR-SVG-20000802/>)

Wood, D. (1992) *The Power of Maps*. Guilford Press, New York.

Zaslavsky, I., Marciano, R., Gupta, A., Baru, C. (2000) "XML-based Spatial Data Mediation Infrastructure for Global Interoperability." Proceedings of the 4th Global Spatial Data Infrastructure Conference, Cape Town, South Africa, 13-15 March 2000. (online at <http://www.npaci.edu/DICE/Pubs/>.)

ACKNOWLEDGEMENTS

I would like to thank Richard Marciano, Amarnath Gupta, Chaitanya Baru, Bertram Ludäscher, Reagan Moore and other members of the Data Intensive Computing Environments groups of the San Diego Supercomputer Center for useful discussions. I am also grateful to David Cleveland, president of Telesis, for providing data and support throughout this project, and to Kathleen Baker for experimenting with AxioMap and commenting on an earlier version of this paper.