

# **BIOGRAPHICAL INFORMATION**

## **Roger D. Harwell**

Product Manager, GeoMedia WebMap Products  
Intergraph Mapping and Geospatial Solutions

### **Specific Responsibilities**

Having joined Intergraph in 1984, Roger Harwell is the product manager of the GeoMedia WebMap products overseeing the “production” of that software from concept to finalized product. During his tenure of over 20 years in the mapping and GIS field, he has managed the development of several products as well as the Team GeoMedia program.

### **Educational Information**

B.S. – Mathematics, Computer Science – Lipscomb University

# Design Considerations of a Web Mapping Application

**Roger D. Harwell**

Intergraph Mapping and Geospatial Solutions  
PO BOX 6695  
HUNTSVILLE AL 35824-0695

## **Abstract**

Many organizations are interested in standing up a web mapping site either for internal or external usage. There are many different reasons these organizations need to do this from simply providing a map picture of an area of interest to performing sophisticated geospatial analysis over the web. A modern geospatially-enabled web site designer will need to take under consideration a number of issues when beginning to design such a site. Its design will lead to decisions in most of these areas.

This paper will present some of these leading issues that need to be addressed. Topics that will be discussed and evaluated include:

- web mapping software: commercial, open source, or none
- platform considerations (operating system, web server software, development environment)
- interoperability considerations including use of Open Geospatial Consortium web mapping standards
- static vs. live data
- map format to deliver to the client (will client-side software be required); advantages and disadvantages thereof
- scalability considerations
- developing a custom site vs. using a visual authoring tool
- what level of geospatial analysis capabilities will be needed
- should web services be employed

Each of these will be discussed, respectively, in the following sections.

## **Software**

One of the first considerations when designing a geospatially-enabled web site will be to determine what software, if any, one is going to use. An obvious pre-requisite is a fundamental understanding of one's requirements for the web site and its capabilities. Then, one can go about evaluating the various software packages to determine which will meet the requirements.

There are four major geospatial software vendors who offer commercial packages:

- Autodesk – MapGuide
- Environmental Systems Research Institute – ArcIMS
- Intergraph – GeoMedia WebMap
- MapInfo – MapXtreme

In addition to evaluating these COTS software products, there are a number of smaller vendors who have such packages and/or are willing to custom implement a site. Likewise these large vendors and/or their business partners are usually willing to engage in custom implementation, though it would typically be on top of their commercial product.

Of course, custom implementation is always an option for the organization—hiring the properly educated and experienced resources and building the site “from scratch”.

A similar option gaining a lot of popularity is to use an open source solution. A number of these have come on the scene in the last few years, including the University of Minnesota MapServer, and others.

## **Platform**

There are many issues to consider with respect to platform when designing a web site. A major consideration will be the decision above (i.e., what software to use) and whether or not that dictates any platform requirements.

An obvious first question in this area is what operating system one intends to use. One may be neutral entering this decision or may have pre-determined reasons for using Windows, UNIX, or LINUX. These reasons could include implementation staff familiar with development tools that only work on one of the operating systems. For example, if one were going to use Microsoft tools like Visual Studio, obviously a Windows web server would be more appropriate. Likewise, if one were going to use J2EE, one might lean toward a UNIX or LINUX operating system, though there are J2EE environments for Windows. There are other operating system considerations, as well. For example, some might consider security to be an issue more with some operating systems than others.

Another consideration is what web server software will be employed. Again, this might be dictated by the software chosen, or by the operating system chosen. But, if there were some reason why a particular web server software package needed to be user, that decision might dictate the former two. The primary commercial web server software vendors and their products are:

- Microsoft – Internet Information Server (IIS)
- Oracle – Oracle *iAS*
- IBM – WebSphere
- Sun – Sun One Web Server
- BEA – BEA Logic

As in the geospatial software decision, open source is also a choice here with Apache being the primary possibility. In fact, most of the commercial products mentioned above are based on Apache.

## **Interoperability**

One key consideration is whether or not there are interoperability requirements of the site. This can be on the reception or serving side. That is to say, there may be a requirement to

interoperate with reading of data provided by others. There may also be a requirement to provide interoperable data to others.

If the requirement is to be able to access other data (departments, formats, servers), one would ensure that his implementation and the GIS software he used were capable of reading from disparate data sources. This might be a proprietary means of accomplishing this, or it might be using third party software, or it might be taking advantage of OGC (Open Geospatial Consortium) standards.

On the sending end, similar questions must be asked. Does the data being exposed from my site need to be shared? Does it need to be exposed as an OGC web service? Again, if these are part of the requirements, the appropriate solution must be sought or implemented.

### **Static vs. Live Data**

Many web mapping sites provide back answers to the clients' questions with data that had been converted to a format accessible by the web mapping software. This means the client is seeing a snapshot of the way the data existed at some point in time.

Other sites are using web mapping products that allow them to present an answer to the client using the latest, "live" copy of their data. If this is a requirement, then such software must be used or a custom solution implemented.

### **Delivered Map Format**

Another important design decision to make is the format of the delivered map to the client's internet browser. While a number of geospatial web software vendors have supported generation of a vector map, some additional typically proprietary software is required to be loaded (either installed on the end client machine or loaded at run-time as in the case of a Java™ applet) in the web browser. As a result, most web mapping software applications are still designed and built to generate raster maps (e.g., GIF, JPEG, BMP, or PNG) which the browsers support rendering natively. When I type in my address at MapQuest or MSN MapPoint, I receive back a raster map in GIF format.

One vector format gaining a lot of popularity is SVG. Scalable Vector Graphics, or SVG, is an industry standard vector graphics format or language. It is an XML encoding or grammar for describing rich, stylable, two-dimensional graphics<sup>1</sup>. It includes advanced graphical features such as transparency, arbitrary geometry, filter effects (shadows, lighting effects, etc.), scripting, and animation<sup>2</sup>. SVG 1.1 is currently a World Wide Web Consortium (W3C) recommendation and on what most implementations is based.

---

<sup>1</sup> See "Scalable Vector Graphics (SVG): XML Graphics for the Web" at <http://www.w3c.org/Graphics/SVG/>.

<sup>2</sup> See <http://xml.apache.org/batik/#SVGSpecification>.

At this time, rendering of SVG also requires additional software beyond what the browsers give you. But with SVG being more of an industry standard, one would hope that the browser vendors would begin supporting native rendering of the format in the not-too-distant future.

Nevertheless, there are a number of advantages and reasons to use a vector map, and specifically to use SVG. One of the main reasons is that you are then working with a much more “intelligent map”, a “smart map”. Rather than just a picture of a map (i.e., bytes designating a color to paint every pixel), there is actual intelligence at the feature instance level. This makes it much easier to work with, for example, a segment of Main Street. There is intelligence within the map that this linear segment extends from one particular X,Y point location to another. Typically, a transformation is understood that allows a mapping from those X,Y points to real-world projected or geographic coordinates making it easier for the client to access that information. Further, some pointer is typically stored with the vector element that provides an association to even further intelligent information. All of these features combine to provide a very intelligent, interactive user experience. As in the example sited, the user can hover his mouse over a segment of Main Street, it highlights to indicate it is the current active element, and a tool tip indicates that it is “Main Street”. The user clicks the element to select it, having had a visual indication that a hot spot is associated with it, and a more detailed attribute report is generated, and/or a video is spawned showing a drive through of the street, and/or insert any creative idea you have on additional information the user might want or need to be associated with this real world feature.

The rendering of the map is certainly dependent upon the rendering software used, but in general can produce a very rich, modern depiction of the area in question. With a vector map, there is much less degradation as you zoom in (without regenerating the map on the server side). Further, with SVG, the ability exists to produce a very rich rendering with features like translucent area fills, patterned area fills, graduated area fills, animation, shadowing, lighting effects, etc.

Some of the mainline geospatial web application providers have the capability built in to generate output formats in SVG. Others have add-ons provided by third-party suppliers to allow generation of the output map in SVG format.

If SVG or some other vector format is a requirement for the site in question, appropriate software would need to be used.

## **Scalability Considerations**

One of the most difficult up-front design issues is trying to determine how many users will be accessing the site during a certain period of time (e.g., daily, monthly, annually) and at one time. Properly designing the web application to *scale* to meet high demand usage is obviously very important. For example, a state Department of Transportation web application may need to typically handle only a few hundred hits per day. But, if they are providing hurricane evacuation route information, there may be times when they have to handle thousands to hundreds of thousands of hits. Therefore an application must be designed to be able to ramp up to handling such high demand usage times.

Scalability considerations may be divided into two areas—hardware and software.

On the hardware side, if one desires his site to be as scalable as possible, he may consider whether a single machine is appropriate or multiple machines acting as a “web farm” typically implemented with some hardware or software-based load-balancing solution. Even each of those machines must have the proper specifications in terms of CPU (speed and number), memory, disk space, etc.

On the software side, if one uses an existing product, will it scale to meet user demand as number of users increases at a particular time or over time? Another side to scalability, is will the software scale as one increases resources, i.e., when one increases RAM, can it take advantage of it? When one adds a second machine, can it take advantage of it? Or will one have to totally or partially redesign the system, re-implement portions?

In a similar vein, is 24-7 up-time necessary? If so, one would typically design a backup server into the system configuration to ensure that if the primary system goes offline, a secondary one can come online.

### **Web Mapping Visual Authoring Tool**

All of the web mapping products contain an API and programmable objects for implementing a custom site. However, in the last couple of years, these same products have been evolving to include more and more tools that prevent the builder of the site from having to be a web programmer. These tools have more and more visual authoring capabilities.

As one would expect, the more highly evolved tool gives less flexibility than building a web application “from scratch”. The tool provider has made some decisions on what the user’s web application will do and look like. For many users, this tradeoff may well be worthwhile in preventing the user from needing the expertise to build and maintain web applications at the programming level.

For some, this may be a major consideration when choosing which product or approach to take in creating their site.

### **Geospatial Analysis**

Part of the requirements definition process will be determining what geospatial capabilities are needed. If a simple map request query is all that is needed, all of the products will be able to meet that requirement.

However, if more sophisticated spatial analysis capabilities are needed, research will have to take place into which products support needed functionality. These might include spatial intersection, difference, or union queries, buffer zoning, linear routing, dynamic segmentation, or label conflict detection.

As the various products have evolved, more and more sophisticated spatial analytical capabilities have become standard features of the product allowing web application developers to provide more and more sophisticated spatial analytical capabilities to their end clients. The ability to ask such questions of a web mapping application is rapidly becoming commonplace.

## **Web Services**

A recent trend in the web industry has been to provide more and more capabilities over the web via a web services model. Web services typically use XML to provide a truly interoperable programming model over the web. This means that an application can take advantage of a service that has been exposed by another server (whether that is another department within the same organization or another organization such as a government agency) across town or half-way across the world.

In the section above on “Interoperability” discussion takes place on a decision of whether or not one’s data needs to be shared. This would be a reason to create a web service providing consumers of the service with one’s data and/or geospatial functionality in a truly system-independent fashion. Another reason to provide and/or use web services is to distribute the processing over disparate systems.

One must determine in designing his site, whether he needs to take advantage of such services and/or expose such services. If either or both are true, an appropriate software solution must be used or implemented.

## **Conclusion**

There are many design decision points that must be traversed when building a web mapping application. This paper has covered some of the more important questions that must be answered in doing said design. It has not attempted to provide all the answers but rather broach the appropriate questions.