

# Web mapping interoperability in practice, a Java approach guided by the OpenGis Web Map Server Interface Specification<sup>1</sup>

P. Fernández, R. Béjar, M.A. Latre, J. Valiño, J.A. Bañares, P.R. Muro-Medrano<sup>2</sup>

Department of Computer Science and Systems Engineering  
University of Zaragoza  
María de Luna 3, 50015 Zaragoza (Spain)  
<http://iaaa.cps.unizar.es>

## Abstract

The Web Map Server Interface Specification, launched by the OpenGIS Consortium in the final part of the year 1999, proposes a standard framework for publish and interchange geographical information, increasing the possibilities of inter-operational capacity among GIS applications in the Internet. On the other hand, software implementation for the internet is highly conditioned by the growth and standardisation of Java as preferred Internet programming language. The authors illustrate an approach for the implementation of the web map server interface using Java as the programming language, in both in the server side to implement the map services of the specification, as well as in the client side, using Java and the more broaden HTML. The paper provides some inside of this experience and shows different examples of map clients accessing map services from one or more map servers fulfilling the OpenGIS specification.

## 1. Introduction

The OpenGIS Consortium (OGC) [2][3][4] is a consensus-based association, which envisions the full integration and interoperability of geospatial data and geoprocessing resources. Recently, pushed by the Internet impact, the consortium has put a big interest in taking advantages of the possibilities open by the web. The following paragraph shows its vision: *"Much geospatial data is available on the web and in off-line archives, but it is complex, heterogeneous, and incompatible.[...] Common interfaces are the only way to enable overlays and combinations of complex and essentially different kinds of geographic information to happen automatically over the Internet, despite differences in the underlying GIS software systems [...]"*.

In the need of a web mapping technology able to manage and divulge geospatial information, we decided to reuse our Java based GIS visualisation technology to develop a web map server, which is adapted to the actual needs of the map&web market and capable to additionally incorporate the new technological advances of this rapidly evolving sector [9][10][11][12]. Because our concordance with the OGC goals on this subject, the OpenGIS Web Map Server Interface Implementation Specification [1] has been the guide.

---

<sup>1</sup> This work has been partially supported by the Commission for Science and Technology of Spain through the project TIC98-0587.

<sup>2</sup> Corresponding author, [prmuro@posta.unizar.es](mailto:prmuro@posta.unizar.es)

The Java programming language has gained quick and large acceptance as a prominent programming language in recent years. Its programming capabilities, multi-platform implementations, facilities for developing distributed applications with powerful built-in high level mechanisms of communication, its powerful standard libraries and the extensive availability of public domain software have converted Java into one of the most suitable tools for software development. In our days Internet and the web are, with no doubt, the preferred medium for sharing information and one of the most powerful mechanisms to provide access to public information where user interactivity is required in some way. In addition to the features explained above, one of the reasons for the Java impact in the programming community has been its ability to integrate with the web.

Some previous work on GIS software projects led us to make the decision of developing our own GIS technology based on the Java programming language platform. On the other hand, not all the work had to be done from scratch because we were able to reuse most of the Java code developed for several, non Internet enabled, previous GIS projects [6][7][8] (this has been very useful to amortize our hazardous initial investment). The nature of the Java platform makes it possible that the same Java GIS kernel supports the needed Web Mapping technology in three different zones of use. First, the Web Server module exploits the GIS kernel to support the map layer displaying capabilities fulfilling the OpenGIS requirements. The client part, also supported with a reduced instance of the GIS kernel, is a Java applet capable of interoperating with the map server. The third part, not so interesting technically speaking, but essential for the end users, is the module in charge of defining the map layers by the users. This last module, based also on the mentioned Java GIS kernel, makes it easier for users to import, manage, render and order graphically their geospatial information, and configuration.

The rest of the paper is structured as follows. First a quick overview of the possibilities of web map services is given in section 2. Section 3 briefly explains the OpenGIS Web Map Server Interface Specification, whereas section 4 deals with a Java approach to develop that specification. Finally, the section 5 illustrates some conclusions and future work.

## **2. Problem overview**

Consider a common situation in the real world. An organization is involved in the study of a certain geographical zone to generate a set of maps. It has some staff collaborating in the task, divided into several work groups. A group is specialized in fieldwork, and they are the best ones recognizing and measuring elements of the environment with a GPS. The second group takes the data provided by the first one and translates it to computer support with a GIS tool. The third one, using satellite images, is in charge of building vector coverages with the geographical elements detected on the images, like rivers, crops, or water reservoirs. Then, the geographers of our organization will construct the maps. They need the data generated by the other groups: the satellite images, older maps, still being used by other members, and other geographic data that an external business provide our organization, like location and characteristics of towns and cities. Finally a computer consultant will publish some nice maps on the Internet, and the generated coverages in the intranet of the organization for internal use.

The problem does not seem too difficult. They only need to be well organized and to follow the planned schedule, but this ideal situation is not likely to take place in the real world. Imagine for instance that the GPS takes points in geographical coordinates, so the

coverages constructed with the GPS points, are expressed in the same reference system and stored with ArcInfo. The satellite images are stored in JPEG format to save disk space, and the coverages created with them are done with MapInfo. To make more difficult the work, the old maps needed were written in DXF format with Autocad, or even as a set of tabular data, and nobody knows in which reference system, scale or date. These are some problems we can find in our work as geographers, but the real matter arise when we need to incorporate any kind of external data. The first and not trivial question is to find the external data we desire, where are the providers, what they are offering, the precision of the data, the scale or date, the prize, or even the maintenance of it. When a provider is selected, the data probably needs to be translated to our coordinate system, and file format, with a special tool. Then the maps can be generated and transferred to the computer gurus to publish them.

A lot of geographical information is available on the web in off-line archives, but it is complex, heterogeneous and incompatible in most of the cases. The OGC is looking for a common set of interfaces to enable the interchange of geospatial data. This is the only way to avoid the actual chaotic organization of the geographical data sources. These interfaces provide a common way to call for data and services facilitating the interoperability between different data repositories and GIS packages.

Serving maps on the Internet is a topic of research nowadays. The interest of users in publishing geographical information on the net, in addition to the ability to get and display information from different sources, have made businesses to make a special effort to cover the clients' necessities. The WMS proposed by OpenGIS and developed with the collaboration of several companies answers to the exposed requirements. The interface specifies a set of services designed to allow the request of maps on the Internet. Commercial OpenGIS servers must implement this interface in order to enable clients, or even other servers, to interchange information. A work environment that needs daily updated geographical information is the best site to introduce a central repository of data accessible through a WMS. The possibility of accessing to several central repositories that offer a WMS interface makes it easier for users to work with local information combined with the last updated data from different catalogues.

The OpenGIS specification of web interfaces, addressed to the most chaotic information source ever known, is a great step to allow the interoperation of multiple users of geographic information along the world. Central repositories of data with common interfaces are one of the main topics of OpenGIS. Clients will use WMS services to obtain information from a central repository, or even from several ones. Cooperation is only possible if the interfaces to access the resources are well known. The web map servers are designed to interoperate and to allow the sharing of geographical information among users. By example, a geographer from the mining department wants to display dynamically a map of a mine with satellite images of the institute of image processing, the level curves of the terrain he has, and the artesian wells from the hydrological department. If all the data sources have the information accessible through a web server offering the OpenGIS WMS interface, our geographer will be able to generate and navigate over the dynamically created map, without having to download the individual files with the data, and having always the latest version of the data.

### **3. The OpenGIS Web Map Server Interface Specification**

The web map server (WMS) interface proposed by OpenGIS is a common set of possible requests that a server is able to answer. As its name properly explains, it is an application

running in a web server that can generate maps to fulfill a specific request with a certain number of known parameters like spatial reference system, format of the answer, size, geographic zone, and the information the client wants. The interface is composed by a set of service descriptions related with serving maps on the Internet. Each implementation must follow the services, methods, conventions and naming that the interface proposes, providing the functions needed to reach the singular objectives of each section of the interface.

The web map server specification has three main kinds of requests: map, feature info, and capabilities. Map requests are sent to the server in order to obtain a specific map file covering a zone of interest; some other parameters, like format of the answer file, reference system, or the information to compose the map, are needed. The feature info requests are an extension of the map requests. Selecting a pixel on the map, the client informs the map server that he wants information about the elements located under that pixel. The WMS will return the information as a text or XML file. Finally a capabilities request refers to the characteristics of the map server. On the response to this request, the map server reports all the services it can perform, the available map data, and the formats and reference systems supported for the answers.

A map request is the main and basic action for a WMS. Its purpose is to obtain a map defined through parameters specified in the request, like zone or content of the map. A map request has also some additional parameters, like required spatial reference system, bounding box of the desired zone, or background color and transparency for image requests. Maps are available in multiple formats divided into three categories depending on the displaying capabilities of clients. Maps can be produced as pictures, series of graphical elements or packaged data. Thin clients will probably ask for rendered maps, directly suitable for displaying in the screen. In this case, the format of the map, specified in the request parameters, will be a picture in a standard format like JPEG or GIF, or a more sophisticated vector graphics format like SVG. When an application has render capabilities, it can choose to ask the server for maps composed only by geographical vector elements in order to visualize them with its own displaying options.

#### **4. A Java approach to the WMS**

Java is the most suitable language to develop applications in Internet, though it is not limited to them because it is a well-designed, object oriented, general purpose, programming language. It also has a large number of standard libraries for many generic and specific purposes so it can be applied to many different types of applications.

Nowadays, our main interest is building "small" information systems with two main characteristics: distribution of the information, with two or more computers sharing data through a network, and integration of geographical information in the system. For instance, mobile fleets tracking systems or natural resources management. These systems require sharing geographical information: maps of the cities where mobiles can be found, dynamically tracked routes, or the real position of cars and lorries.

Taking profit of our knowledge of GIS and the functionality these systems require, two years ago we decided to develop a GIS visualization kernel to support the needs of our information systems. This way we had the flexibility that commercial GIS visualization libraries did not offered us. Java was the programming language selected to develop this kernel because of its inherent distributed capabilities, multi-platform support and Internet-orientation, besides it is a well-constructed object oriented language.

The web map server presented in this paper and all of the complementary tools that need GIS visualization, are built around our GIS kernel. This module has also been successfully integrated into other projects, like a mining resources management system, GPS fleet tracking systems, or handling and visualization of satellite images (see [5][6][7]).

#### 4.1. Web Map Server software architecture

Our web map server is made in Java and uses the geographical data management and rendering capabilities of our GIS kernel to generate and display maps. It adds the functionality needed in order to fulfill the OpenGIS interfaces. It consists of four applications, as can be seen in Fig. 1. The JMapServer is the main application. It is in charge of receiving the map, capabilities or feature information requests, generating the answer as a file, and returning it to the requester.

The JMapServer offers a Java RMI interface with the same services described by the OpenGIS web map server. RMI is a standard mechanism that Java offers to define and communicate remote objects coded in this language [13]. The server has three different modules: the GIS visualization module is the GIS kernel mentioned before, and it is in charge of managing and displaying the maps. The second module, the MapBuilder, receives the requests and creates files with the answer maps using the visualization module. This files are the data returned to clients. Feature information requests are also responsibility of this module. The capabilities module complements the JMapServer. At initialization time, it loads and parses the capabilities file and configures the JMapServer. When a capabilities request arrives, the map request builder will use it to reply with the actual capabilities.

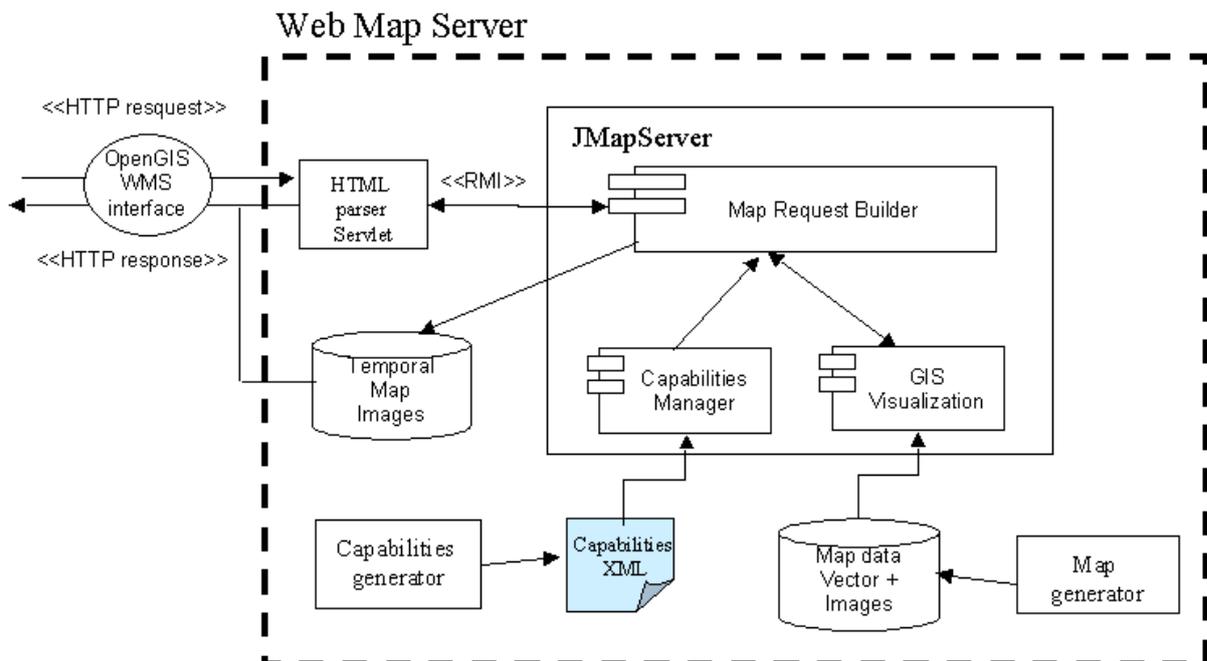


Figure 1: Web Map Server modules and configuration tools

The JMapServer generates response maps given the requests, but it is not directly accessible through the net. A support component is used to parse the HTTP requests received

by the web server that are destined to the map server. A Java servlet [15] is used for this purpose. It is a simple Java class installed in the web server that collects, translates and sends the map requests to the JMapServer, using its RMI interface. The JMapServer does not receive the requests directly as an HTTP request, needing thus the intermediate servlet, because this way we can separate the functionality offered by our JMapServer and the OpenGIS interfaces. Besides this, the JMapServer can be accessed through its RMI interface by any other Java program. This architecture makes it easier building remote client applications in Java.

#### ***4.2. Web Map Server Configuration Tools***

For managing the JMapServer and the geospatial information it has, we have developed two additional applications that are part of the web map server (see Fig 1). The WMS capabilities, as described in the OpenGIS specification, are defined in a configuration file, written in XML. As these files are too complex for a user to manually create and update them, we have developed the capabilities generator, that it is an application that provides a graphical user interface to configure this file. The web map server reads the capabilities file at initialization time, in order to configure itself, and the same capabilities are returned in response to capabilities requests.

The other application is a map visualization tool, that allows to generate the maps that will be served by the map server. This application is also based in the same GIS kernel than the map server. This is a clear proof of the reusability and extensibility of our GIS kernel, qualities that were achieved thanks to a good design and the use of the Java language. Exactly the same application can be used by remote clients to interact with the system, merging local data with the maps served by our map server, or by any other OpenGIS compliant map server in the Internet.

#### ***4.3. Web Map Clients***

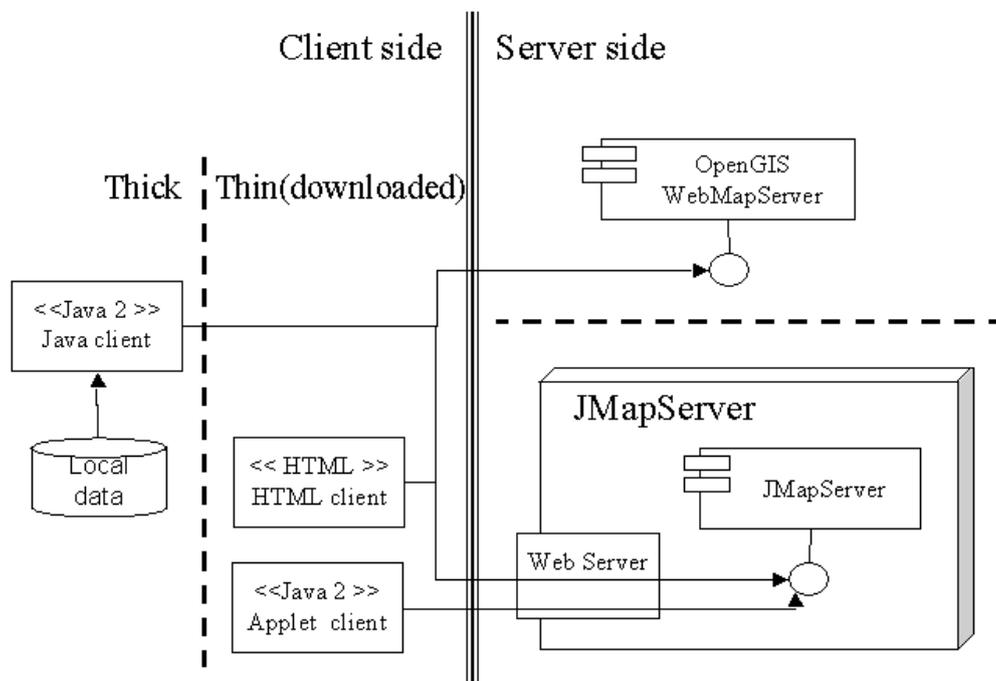
We have developed three types of clients suited to different users' necessities (see Fig 2), that constitute the basis to build more complex applications that need to access to one or more OpenGIS web map servers. The three client applications allow basically the same operations, as showing the map on the screen, selecting what layers will compose the map and navigating over the map, but the flexibility and power of these tools are different in the clients.

An HTML page is the simplest client. It's a light page that access to the map server. It has a small download time and it is simple enough to allow home users to see some nice maps. The page shows a map and presents some basic tools that allow users to interact with the map changing the visible zone or the scale, and selecting the visibility of several predefined layers defined in the map server. It is also possible to get image maps from different map servers, and to compose them in the HTML page.

The second client is a Java applet that accesses the WMS. This applet is a small version of the same GIS module used in the server to generate maps. It implements the same map tools than the HTML page, like zoom or pan, and adds others, like the display of the coordinates of the mouse over the map, the possibility of changing the scale of the map, or a distance measurer. The applet has also a selectable list of the layers and styles available in the map server. The advantages of the Java applet are the increase of functionality and the flexibility in the access to the data because all of the parameters are obtained dynamically from the map

server, while several of them, as the available layers, are fixed in the HTML page. The applet is downloaded from the computer with the web map server. In other way the applet would not work because of Java security restrictions that don't allow the applet to connect to servers different from the one where it was downloaded. This strong restriction avoids that Java programs could do any action not desired by the user, like damaging the system or broadcasting confidential information. To share information from several sources, the HTML page or the local application could be used. To use the applet to obtain data from several map servers, the browser user has to give permission to the Java applet to access other servers.

The Java applet of the previous point is a simplification of the general visualization tool used to generate map descriptions for the map server. This visualisation application can also be installed in a user's computer to access to the web map server, in the same way as the applet. Local deployment also carries improvements in the functionality, overall addressed to specialized users. The user now can access to several OpenGIS web map servers, as well as to local files. This user can work with both kinds of data in the same way.



**Figure 2: Web Map Server architecture and clients configuration**

In figure 3 you can see a picture of the HTML map client. The web page has a central map, with several selectable layers of data on the right part. Below the map are tools like zoom and pan. If one of these tools is selected clicking over the map will perform the operation, making a request to the WMS with the options selected by the user. The Java applet (see Fig 4) has basically the same tools. It can perform a zoom, a pan or a selection dragging with the mouse over the map. It shows the last map answered by the map server. The content of the request, layers and styles, can be selected from the tree structure in the left part. It shows the layers and styles structure of the map server that has been obtained via a capabilities request to the server. The general visualization client has the same structure, but a different layer legend is shown for each map server where the user wants to request data. Also, in the same way, local data can be loaded and merged with the remote information.

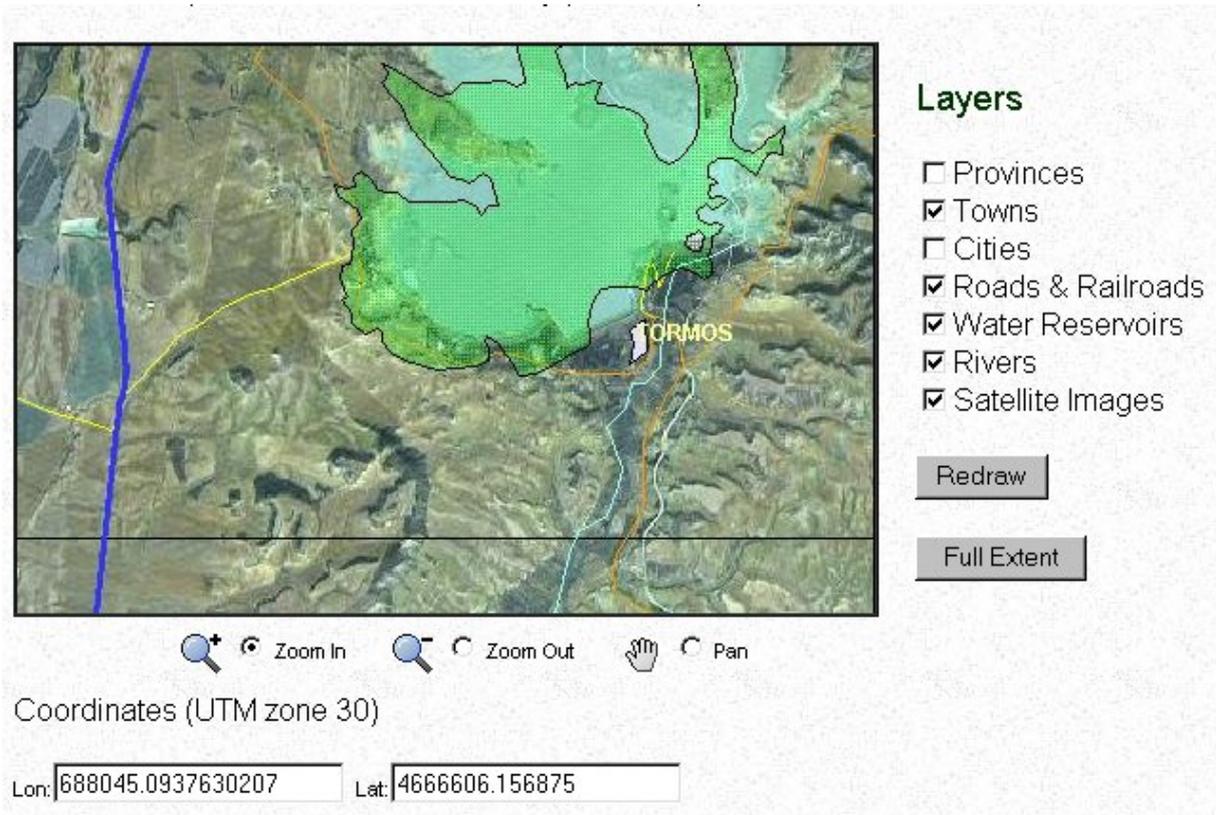
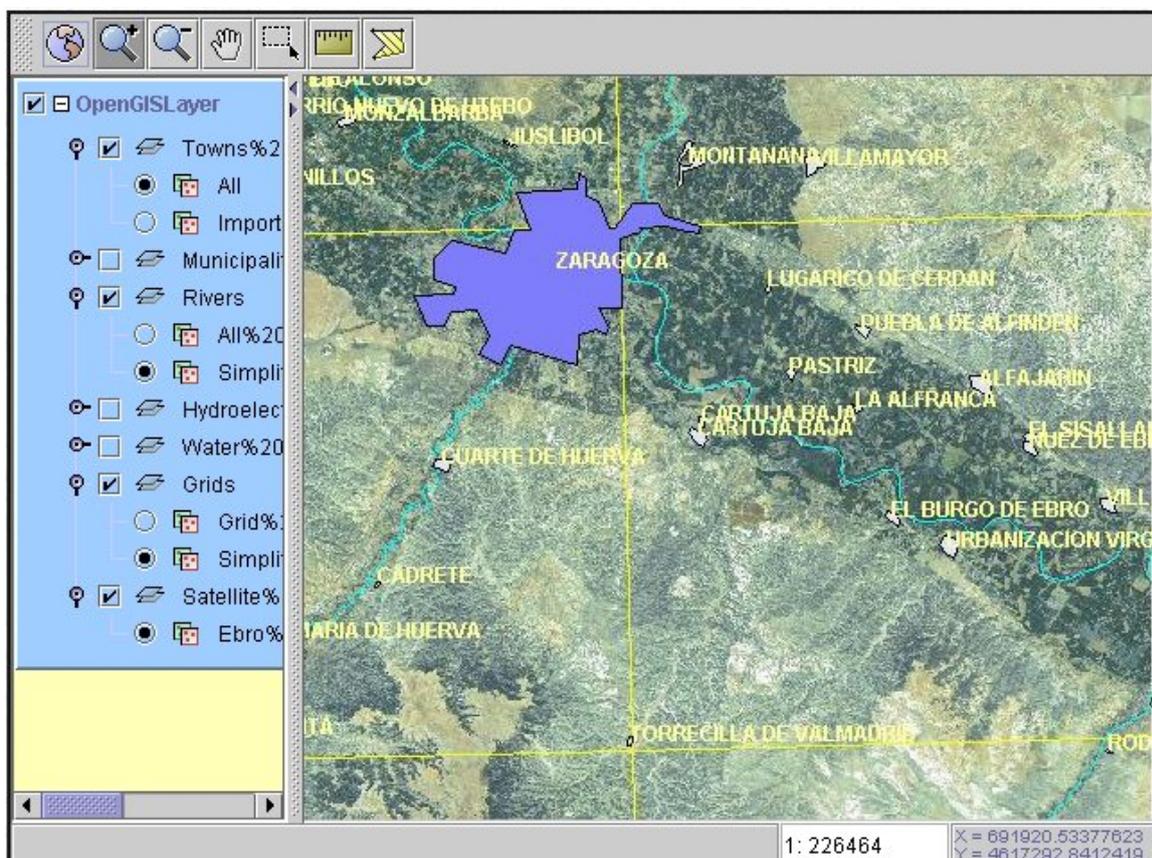


Figure 3: HTML map client



## **Figure 4: Java2 Map Client**

### **5. Conclusions**

The WMS can manage a central repository of maps accessible from several sources and without the complexity of some commercial systems, but it can be more than a tool to share data. The software developed in the project is only a base to build more complex distributed geographic information systems, where a powerful commercial GIS system can be too large, difficult to use, and expensive.

Our map server still needs to be improved in, at least, two ways. First, the server has to be more open. External applications must be able to generate maps on-line, and integrate them into the map server. Adding additional interfaces in RMI or CORBA, to the map server, can do this. In the current implementation of our WMS, we added an RMI interface to the server with exactly the same services that the OpenGIS one (see Fig 1). The next phase will consist in adding new services to this interface to address online management of information (a child work for us ☺). Thus real time actualization of data can be performed by external applications, for example, a fleet tracking system can modify the positions of the mobiles, so that the next maps built by the map server will have always the latest information.

The access to distributed geographic information is solved in different ways by the previous tools, so now is time to make the best use of Java, and reuse the implemented services for accessing the web map server in more complex clients<sup>3</sup>. A representative example of its use can be the GIS visualization module of a fleet management system accessible by the Internet. In this case, while a server module can be in charge of updating the mobile positions when GSM short messages arrive, the client can access the map services to display the latest information coming from the central WMS and combine them with its own local data.

### **6. References**

- [1] OpenGIS Project Document 99-077r4, OpenGIS Consortium 2000 “*OpenGIS Web Map Server Interface Specification (version 1.0)*”.
- [2] OpenGIS Project Document 99-112, OpenGIS Consortium 1999. *The OpenGIS Specification Model. Topic 12: The OpenGIS Service Architecture (version 32)*.
- [3] Homepage del OpenGIS Consortium. <http://www.opengis.org>
- [4] Cliff Kottman. “*Introduction to OpenGIS Consortium, Inc.*”. Presentation in a OpenGIS meeting, Vancouver, Canadá. Feb 7, 2000. Accesible in <http://opengis.org/techno/presentations.htm>.
- [5] John D. Evans. “*The MIT OrthoServer and Interoperable Web Mapping*”. GIS Colloquium Lecture Series. MIT Dept. of Urban Studies and Planning. Mar. 2000. <http://icg.fas.harvard.edu/~maps/hgis/Ingis.htm>.
- [6] P. Fernández, J. Nogueras, O. Cantán, J. Zarazaga, P.R. Muro-Medrano. “*Java Application Architectures to Facilitate Public Access to Large Remote Sensed and*

---

<sup>3</sup> To see the results of our works you can visit: <http://iaaa.cps.unizar.es/demos/demos.html>

- Vector Geographic Data*". Telegeo '2000. Second International Symposium on Telegeoprocessing. Sophia Antipolis, pp. 81-92. France, 10 –12 May, 2000.
- [7] M. Á. Latre, R. Béjar, P. Fernández, P. Álvarez, P. R. Muro-Medrano. "*Trying Java technology in a Geologic-Mining Information System distributed over an inter/intranet environment*". Telegeo '2000. Second International Symposium on Telegeoprocessing. Sophia Antipolis, pp. 93-102. France, 10 –12 May, 2000.
- [8] F.J.Zarazaga, P. Álvarez, J. Guillo, R. López, J. Valiño, P.R. Muro-Medrano. "*Use Cases of vehicle location systems based on distributed real-time GPS data*". Telegeo '2000. Second International Symposium on Telegeoprocessing. Sophia Antipolis, pp. 53-62. France, 10 –12 May 2000.
- [9] W.F. Limp, "*WEB MAPPING*". GEOEurope. N. 8, pp. 18—22. Dec. 1999.
- [10] A. Sorokine, I. Merzliakova. "*Interactive map applet for illustrative purposes*". Proceedings of the 6th International Symposium on Advances in Geographic Information Systems. pp. 46—51. 1998.
- [11] Yafang Su, Joan Slottow, Avi Mozes. "*Distributing proprietary geographic data on the World Wide Web—UCLA GIS Database and Map Server*", Computer&Geosciences N.26 pp.741-749, 2000
- [12] Harder C, "*Serving maps on the Internet*" Environmental Systems Research Institute Inc, Redlands, California, 130 pp. 1998
- [13] Rober Orfali, Dan Harkey, "*Client/Server Programming with Java and Corba*", second edition, Wiley Computer Publishing, 1998
- [14] Homepage of Java, from Sun Microsystems. <http://www.java.sun.com>
- [15] Bruce Eckel, "*Thinking in Java*", Prentice Hall, 1998