Dynamic webgis and tracking of mobile devices

Michele Ceccarelli*, Francesco Cioffi*, Michele Di Capua+

*Research Centre on Software Technologies RCOST
University of Sannio, Benevento, Italy

+Unlimited Software s.r.l.
Centro Direzionale, IS F11, Naples, Italy

A webgis is a system to manage and deploy spatial data and associated attributes on the web. Usually, the data are static on the client-side: a standard webgis application doesn't manage dynamic information. In this article we illustrate a dynamic webgis: a system architecture that is able to automatically collect, manage and display dynamic geographic information. In this way one can monitor in real-time on the map the state of time-evolving entities. Possible applications are fleet tracking, environmental control, surveillance. Actually the system is being applied to fleet tracking.

1 Introduction

Every day, people use on-line maps for a variety of uses, from planning road journeys to exploring unknown locations. A Geographic Information System[1, 2] (GIS) is a system for creating and managing spatial data and associated attributes. It is a computer system capable of integrating, storing, editing, analyzing, sharing, and displaying geographically-referenced information.

Web-mapping is now one of the most simple and widely accessible uses of GIS. A standard webgis[8] is able to show geographic information on the web. In a standard webgis, the users can browse the web page through the hypertext linkage. Each web page is a static image organized by the web developer. Static pages are static HTML pages that are prepared in advance of the request. A geographical document from a static webgis is stored as a file and its contents are determined (and fixed) by its author when it is created. The GIS server returns the HTML pages to the user (as shown in fig. 1).

This information is static: once the client has requested and received data, these data can change on the client, if and only if, the client send a new request to the server (common HTTP protocol behavior). The main drawback of this kind of architecture is that every time the user wants to change or interrogate the map, the server has to update the view and send it to the user. This can create an impression of poor interactivity, but this architecture is not able to show the state of an entity after it is changed.

In [6] the advantages of webgis dynamic ability are exploited for developing dynamic webgis. Dynamic pages are created in response to a users request. Each time user accesses the site he will view the different information (depending on the request). Webgis is dynamically linked with the sources. This dynamic linkage with the sources always keeps

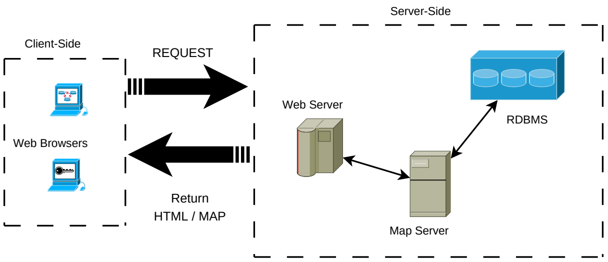


Figure 1: Webgis architecture

the data and software current. On the server side, the CGI script is responsible to provide real-time multimedia information. Webgis can link with multimedia real-time information by real-time connection with the relevant information sources.

[7] uses webgis for monitoring soil erosion in Miyun Reservoir area. A soil erosion monitoring information system that is dynamic, interactive was developed based on the software of ArcIMS. The ArcIMS server software can be implemented as an image server or feature server. An image server serves the data on the server side and creates images in JPEG format, which it sends to the browser client in response to user operations (i.e. pan, zoom, select). Image server mode has been applied in soil erosion monitoring information system.

It is our opinion, that in a dynamic architecture, the changes should be visible, to the user, in real-time. We mean, for dynamic webgis, an environment where the entity state can change many times. If one has to monitor the state of an entity (eg. temperature or alarm sensor in a building), there is the need of managing dynamic information, in a more responsive way. Our aim, here, is the proposal of a dynamic architecture consisting into a real-time webgis where the user can observe changes over the map as the corresponding information is available and updated from external sources, without explicit requests by the user.

With this dynamic application it is possible to view in real-time for example the state of mobile devices and their position also. An example can be a system for fleet tracking in which a user needs “to follow” some auto vehicle, real-time on a map. The main differences with previous approaches is the adoption of an active client that continuously updates information requesting them to the server. The client updates only dynamic layers without refresh other static elements in page.

In the next paragraph we analyze some open source webgis software. We have choosen one of them. The next paragraph describe our application: we illustrate components and architecture, and describe the features that we have developed and integrated in a prototype version. Then we discuss other case studies accessible by the presented architecture and possible extension to be developed.

2 Some open source Web GIS applications

In order to develop the architecture of a dynamic webgis as introduced above, we analyze some of most diffused open source webgis systems which could be used as the starting point for the development of our architecture. Here we briefly present some of the main features of four such applications.

2.1 UMN MapServer

MapServer[9] is an open source development environment for building spatially-enabled Internet applications. MapServer is not a full-featured GIS system. Instead, MapServer excels at rendering spatial data (maps, images, and vector data) for the web.

Beyond browsing GIS data, MapServer allows you create “geographic image maps”, that is, maps that can direct users to content.

2.2 GeoServer

GeoServer[10] is a fully functional open source WFS-T and WMS server that follows the OGC [11] open standards.

With GeoServer it can be possible to publish data as maps / images (using the WMS), as actual data (using the WFS), and allow users to update, delete, and insert features (using the WFS-T). The focus is easy of use and support for open standards, in order to enable anyone to quickly share their geospatial information in an interoperable way.

2.3 MapIt!

MapIt![12] is a server-side web-application for raster-maps. Navigation and points of interests are easily configured.

2.4 Alov / TimeMap

Alov[13] / TimeMap[14, 15] is free, portable Java application for publication of vector and raster maps to Internet and interactive viewing on web browsers. It supports the complex rendering architecture, the unlimited navigation and allows working with multiple layers, thematic maps, hyperlinked features and attribute data.

TimeMap TMJava is a novel mapping applet which generates complete interactive maps with a few simple lines of HTML. It provides a way of easily enriching web pages with historical or contemporary information that goes far beyond static JPEG map images. It's easy for beginners, yet provides completely customizable power and distributed back-end database connectivity for the expert. It's free for personal use.

3 USAlov system

The developed system, termed USAlov, is based on Alov / TimeMap. TimeMap is developed in J2EE[4] technology and it is very flexible and easily extensible. USAlov[3] is a J2EE / J2ME[5] application developed to track locations of remote mobile devices storing their positions on a relational database. It can be used to view real-time locations of mobile devices.

3.1 Components

Our mission has been the integration and development of new features in Alov / TimeMap. We have added a tracking server, and a mobile component, while the applet has been extended with modules for monitor the state of targets.

Tracking server is a new server-side component, that manages dynamic information about mobile devices. It stores this data into a MySQL database, and stores in memory the information about active devices. We have developed a mobile component that runs on mobile devices and communicate, by the bluetooth protocol, with common GPS[18] receivers, and, by GSM network, with the tracking server. Last, we have extended the basic TMJava applet, used to view static GIS data, with the capability to communicate with tracking server and show dynamic information. The extended applet allows to dynamically query the tracking server, to get the position of the mobile devices (see fig. 2).

3.2 Architecture

The dynamic webgis architecture is an extension of standard webgis architecture. We have expanded the client / server standard webgis architecture as client / server / mobile architecture (see fig. 3). The tracking server al-

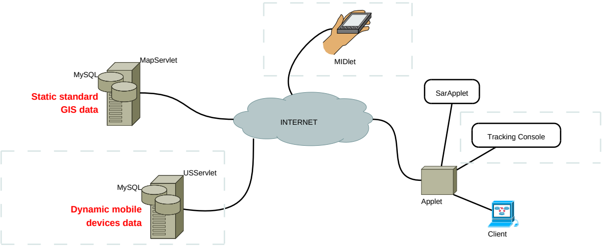


Figure 2: Components

lows communication between mobile and applet. A client requests static data to the map server, as usually happens in all webgis environments, and requests dynamic data to a tracking server. So, the map server works as in standard webgis, it sends geographic information on static elements (eg. maps), to client, while the tracking server provides information about dynamic entities, and at the same time receives information (eg. state) by mobile devices.

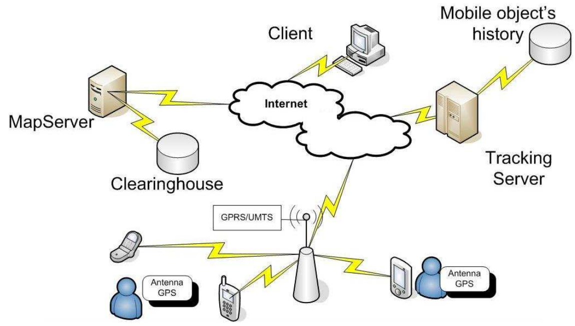


Figure 3: System architecture

The mobile device, J2ME enabled (eg. mobile phone), communicates with a GPS receiver, and it sends information about its location to a web server. Thanks to GPRS[19] / UMTS[20] connection and short size (few bytes) messages it is possible to use this application in a not expensive way.

The server stores the messages from mobile devices into a database and in an active structure that resides in the server. A thread is responsible to load information about monitored targets in memory.

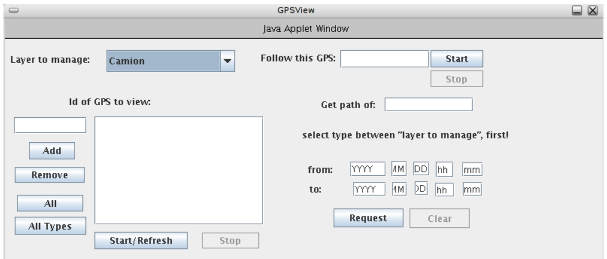

GPSView
Java Applet Window
Layer to manage: Camion
Follow this GPS:
Id of GPS to view:
Get path of:
select type between "layer to manage", first!
from: YYYY MM DD hh mm
to: YYYY MM DD hh mm

Figure 4: Tracking panel

The client is based upon Sun Java applet technology, which allows to view the whereabouts of an entire fleet from a desktop. The applet provides a tracking panel (shown in fig. 4) used to configure the request to send at tracking server. After query, the client runs a thread that updates the requested data at configurable intervals time. Indeed, the applet is in a continuous polling to the server with HTTP requests.

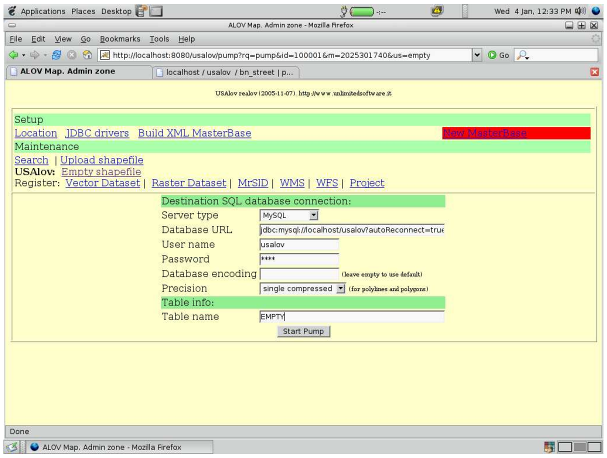
3.3 Database

The server is connected to a database (Clearinghouse) which includes records where the description of GIS data-sets is stored. These records include descriptive data for resource discovery allowing identification of relevant resources to a particular place, time or thematic interest. They also contain connection parameters for each data-set which allow the server to connect to the data servers across the Internet (or to local files).

There is a dedicated WWW interface to upload data from shape[16] or MIF[17] files to a relational database and to register these data in Alov Clearinghouse.

Mobile devices history is stored on different database. Each time that a registered device sends its location, the server stores this data and other related information (timestamps, etc.), on a database.

Another dedicated WWW interface is used to register mobile devices in order to keep a simple (registration required) level of security.


Applications Places Desktop
Wed 4 Jan, 12:33 PM
ALOV Map. Admin zone - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
Go
ALOV Map. Admin zone
localhost / usalov / bn_street | p...
Setup
Location JDBC drivers Build XML MasterBase
New MasterBase
Maintenance
Search | Upload shapefile
USAlov: Empty shapefile
Register: Vector Dataset | Raster Dataset | MrSID | WMS | WFS | Project
Start Pump
Done
ALOV Map. Admin zone - Mozilla Firefox

http://localhost:8080/usalov/pump?rq=pump&id=100001&m=2025301740&us=empty

USAlov realov (2005-11-07). http://www.unlimitedsoftware.it

Destination SQL database connection:

Server type	MySQL
Database URL	jdbc:mysql://localhost/usalov?autoReconnect=true
User name	usalov
Password	****
Database encoding	(leave empty to use default)
Precision	single compressed (for polylines and polygons)

Table info:

Table name	EMPTY

Figure 5: Upload interface

3.4 Features

These are the basic features available at this moment:

- *Fleet Tracking* The system can track mobile devices, filtered by type or by a particular attribute (eg. "id") (see fig. 6).
- *Follow mode* The system can follow a single object on map. The maps scrolls automatically, centering the selected element on the screen.
- *History* Analyze previous devices paths within a given interval time (see fig. 7).

The mobile devices model is based on Java classes. These classes can have different attributes. For example a "car" type can exists with a license plate attribute while a "bus" type can have seats and standing room attributes. The application can manage different custom types.

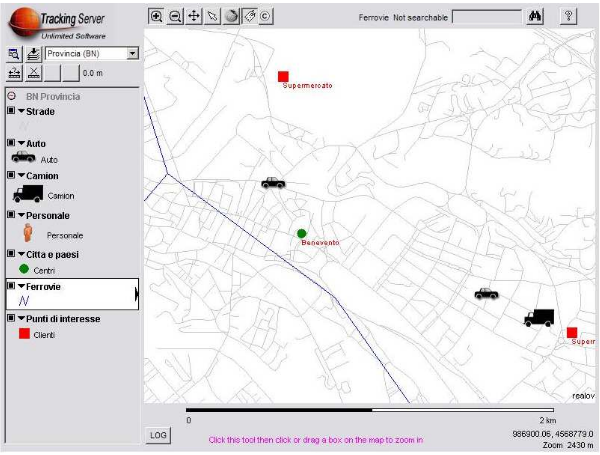


Figure 6: Fleet Tracking

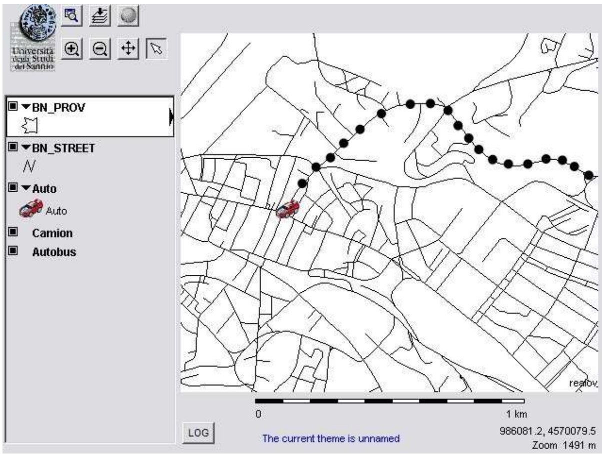
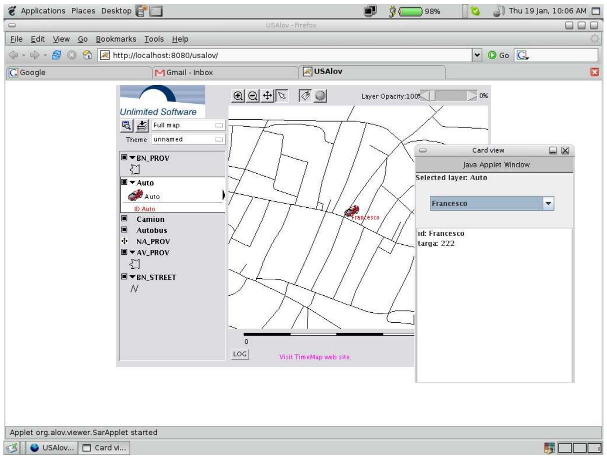


Figure 7: History

4 Case studies

USAlov is only one of the possible extension supported by this architecture. We can imagine different scenarios, where a tracking server can be used.

- *Surveillance agency* A surveillance agency needs to track their agent's path, and it need to know, at any time, their position. Any agent has a device that sends information about its location to a server, so their job is tracked and



http://localhost:8080/usalov/

Full map

Theme unnamed

Selected layer: Auto

Francesco

id: Francesco
targa: 222

Applet org.alov.viewer.SarApplet started

Figure 8: Properties pop-up

stored on database. With a custom mobile application, the agents can send to the main server their activity check information in real-time.

- *Human resource management* With this tool, a company is able to manage better distribution of human resources (eg. technicians) on a localized area, thanks to integrated distance tool.

5 Conclusions and extensions

We presented the architecture of a tracking server with a new and innovative concept of dynamic webgis. The illustrated architecture it also extensible and flexible, and can be adapted to more applications environments. Indeed, the technology used is brand new, and it can be integrated with many other scenarios such as follows.

- *Image server.* An interesting add-on can be the integration of an image server able to generate on demand, graphical information. In this way the system aren't linked to a particular data map and it has not care about GIS data.

- *Communication between mobile devices* A user can receive information about its nearest neighborhood on a configurable distance. For example, the system can detect if a user enters in an area and advice users in the same area.

References

[1] Burrogh P.A. 1986, "Principles of geographical information systems for land resource assessment", Clarendon Press, Oxford, U.K, 194pp.

[2] Mogorovich P., Mussio P. 1988, "Automazione del Sistema Informativo territoriale. Elaborazione Automatica dei Dati Geografici", Masson, ITA, p. 503-8 vol.2

[3] F. Cioffi 2006, "Tracking of mobile devices in webgis environments", Thesis in Computer Engineering, available at http://www.fcioffi.net

[4] Derek C. Ashmore 2004, "The J2EE Architect's Handbook", DVT Press, Lombard, IL, 306pp.

[5] John W. Muchow 2001, Core J2ME Technology & MIDP, Prentice Hall, 737pp.

[6] T. Soomro, Zheng, K. Pan, "HTML and multimedia Web GIS", Proceeding of Third International Conference of Computational Intelligence and Multimedia Applications, 1999. ICCIMA '99., IEEE Press.

[7] Jianxi Huang, Bingfang Wu, Wenbo Xu, Yuemin Zhou, Yichen Tian, Wenbo Huang, "WebGIS for Monitoring Soil Erosion in Miyun Reservoir Area", Proceeding of IEEE International Geoscience and Remote Sensing Symposium, 2004. IGARSS '04., IEEE Press

[8] Tang Dali, Huang Jixian, "Structure and technology on Application-WebGIS", Proceeding of International Conferences on Info-tech and Info-net, ICII 2001 - Beijing. 2001, IEEE Press

[9] http://mapserver.gis.umn.edu/, June 2006

[10] http://docs.codehaus.org/display/GEOS, June 2006

[11] http://www.opengeospatial.org, June 2006

[12] http://www.mapit.de/index.en.html, June 2006

[13] http://www.alov.org, June 2006

[14] http://www.timemap.net, June 2006

[15] Ian Johnson, "Putting Time on the Map: Using TimeMap for Map Animation and Web Delivery", Proceeding of GeoInformatics, 2004

[16] "ESRI Shapefile Technical Description, An ESRI White Paper", 1998

[17] http://www.csh.it/mif_mid.htm, June 2006

[18] B. W. Parkinson and J. J. Spilker, Jr., "GPS signal structure and theoretical performance," in Global Positioning System: Theory and Application, vol. I, Eds. Washington, DC: American Institute of Aeronautics and Astronautics, 1996, ch. 3, pp. 57-120.

[19] http://en.wikipedia.org/wiki/GPRS, June 2006

[20] http://en.wikipedia.org/wiki/UMTS, June 2006