M-GIS — Mobile and Interoperable Access to Geographic Information

Jorge Cardoso[1], Artur Rocha[1], João Correia Lopes[1,2]

[1] INESC Porto, R. Dr. Roberto Frias, 4200-465 Porto
http://www.inescporto.pt/
{jcsc,artur.rocha,jlopes}@inescporto.pt
[2] Faculdade de Engenharia da Universidade do Porto

Abstract. This paper describes an architecture which can be used to access geographic information from mobile devices with limited display and processing characteristics, such as Personal Digital Assistants (PDA) or Mobile Phones. The information may come from different sources leading to an interoperable solution.

M-GIS obtains geographic information from sources described using Geography Markup Language (GML) from one or more *Web Feature Servers* allowing an access independent from its format or physical location, as long as the information is according to the specification.

Using XSLT, GML is transformed to a graphical format in Scalable Vector Graphics (SVG) which can then be manipulated in the mobile device. M-GIS follows a client/server architecture and the client application was developed using *Java Mobile Information Device Profile* technology.

The results enable us to conclude that the mobile system, designed and developed with the use of open standards and representation formats to access geographic information, is a viable solution with some limitations. The main limitations of this approach have to do with the amount of information which the client can, at the moment, accommodate.

1 Introduction

The event of web-enhanced portable devices widens the way for a whole new series of personal and professional applications. It's becoming a common procedure to download utilities or leisure applications to use in one's mobile phone or *Personal Digital Assistants* (PDA). JAVA enabled platforms have given a broad contribute to this *status quo*, due to the one-deployment-fits-all philosophy.

In the Geographic Information Systems (GIS) application field, the potential has been identified: it's a world map in the palm of your hand, wherever you are, whatever you're doing and with no "strings" attached. This is carefully being explored by GIS and GPS companies at the pace that technology allows and strategy advises, having as main goal to achieve the best performance in a specific application for a platform of their choice.

Although this is probably the only way for a commercial approach, it is not the goal of the work presented in this paper. Of course that performance is

desirable and was present on most of the choices during this development, but the main concern of this work was to achieve full interoperability for a cross-platform GIS application in mobile environments.

The requirements are high, but standards proved they are up to the task (several were used in developing M-GIS) and a fully operational prototype was built. Performance is, predictably, its weakest point, but a few ways of improving it are also presented in this paper. On the other hand, the increasing processing capability of mobile devices, allied to broader communication bands sought for 3rd generation mobile networks, may help proving the added-value of solutions based on this prototype, specially for professional applications.

There are some implementations available for PDAs that are able to show the map of some area and may allow the search for locations of particular places or buildings. ArcPad [ESR03] from ESRI, GeoGIS [Geo03] from Geo InSight and PocketGIS [Poc03] from Pocket Systems do not offer interoperable solutions, as they are limited to show information described in a particular format, usually from a particular vendor.

M-GIS — Mobile Geographic Information System [Car03] follows a client/server architecture; it obtains geographic information from sources described in Geography Markup Language (GML) [OGC01], using XSLT transforms it to a graphical format described using *Scalable Vector Graphics* (SVG) [W3C03], which can then be manipulated in the mobile device.

Geographic information in GML can be served by one or more *Web Feature Servers* allowing an access independent from its format or physical location, as long as the information is according to the specification. The client application was developed using J2ME (*Java 2 Micro Edition*) technology, the profile MID (*Mobile Information Device*).

This paper continues by presenting in Section 2 an hypothetical situation, often experienced by work teams in public administration, in which M-GIS could prove of great value; Section 3 presents the M-GIS architecture design and shows a sequence of interactions with the M-GIS client application; Section 4 describes the implementation of M-GIS server and client; Section 5 presents an assessment of the technology used and, finally, conclusions and future work are presented.

2   Problem Description

This section describes an hypothetical situation, but one that is very real and often experienced by work teams in public administration, in which M-GIS could prove of great value. An inspection team for economical activities, "Team E", leaves on their car to perform its daily work. Upon arriving to municipality X, which they did not visit for over 6 months, it urges to request updated information on cadastre, buildings or even roads, for the area they are aiming to inspect.

How to solve this problem? Each municipality will have its own GIS system, even inside the same city hall and sometimes different GIS technologies coexist. Which software is used? What kind of connection is needed?
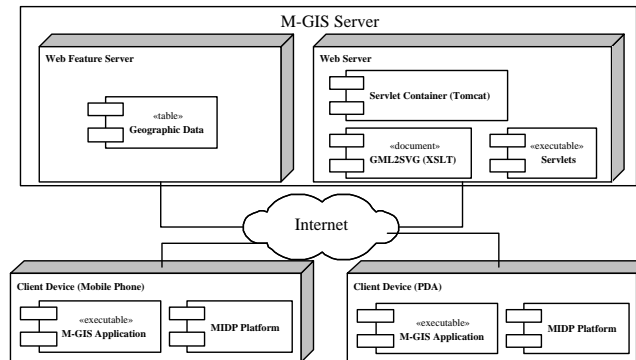
Fig. 1. M-GIS Architecture

Now, another team, also from the public administration, but with a completely different set of responsibilities, decides to visit this same municipality. Will they have the same kind of devices than Team E or a Java enabled mobile phone is sufficient?

This kind of scenario, where interoperability problems arise, lead to the requirements considered for M-GIS.

3 System's Architecture

This section presents the architecture, designed and implemented to tackle the kind of problems sketched before, and describes the usage requirements that are devised for it.

The system's architecture is presented in Figure 1. The M-GIS system is composed by a *Web Server*, a *Web Feature Server* (WFS) and by devices with the M-GIS Client Application.

The geographic information is obtained from a *Web Feature Server* (WFS), which can itself be configured to get its data from other Web Feature Servers. The WFS provides geographic data in GML format which is dynamically transformed into Scalable Vector Graphics (SVG) upon a request by the client M-GIS Application. The transformation from GML to SVG is made using a XSL Transformation (XSLT) [W3C99] and the resulting SVG document, that constitutes the map, is then sent to the client application for displaying.

The M-GIS Server

The server responds to two kinds of requests: list of available layers in the WFS and content of a specific area. The first one is processed by asking the WFS for the list of available layers and passing it to the client. The second kind of requests triggers a more complex process. The area request is passed on to the Web Feature Server (with some syntax modifications) which will return a
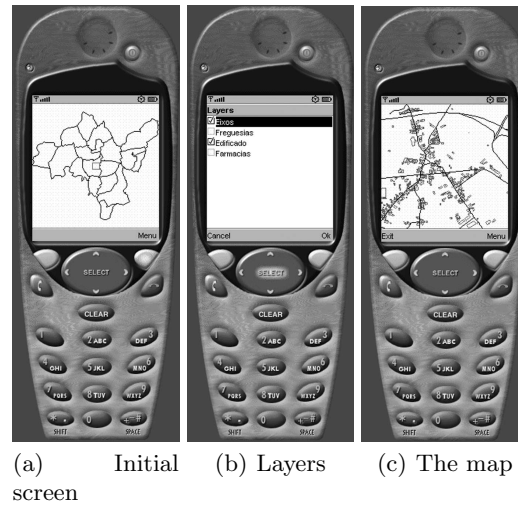
(a)   Initial   (b) Layers   (c) The map
screen

Fig. 2. Interaction sequence

GML document with the geographic data. This GML document will, in turn, be transformed to SVG by the server using a XSL Transformation. Finally, the SVG document will be sent to the client application.

All the communication between the client application and the M-GIS server is made through HTTP, with parameters sent in URL-encoded parameter-value pairs.

M-GIS Client Application

The client application is a Java MIDlet with the following main functionalities: select the area to visualise; pick the map layers to display; navigate the map: pan and zoom the view; search and identify features on the map.

Figure 2 shows displaying and interaction with the M-GIS Client Application. In this application, the user first selects the area to visualise over a pre-loaded map on the device (Figure 2(a)) containing administrative boundaries, in this case, at a local administration level. After the area has been selected, a list of available map layers will be displayed for the user to choose from (Figure 2(b)). Finally, the corresponding map will be displayed and the user will be able to pan, zoom, identify and search for features (Figure 2(c)). Once the map has been loaded into the device, all operations on it are made locally, i.e., there is no need to communicate with the server. The server only needs to be contacted if we want to load a different map.

4   Implementation

The M-GIS server was developed as a set of Java servlets and a XSLT stylesheet. As for the Web Feature Server, the Deegree [Dee03] WFS was used since this is an open source and free software. Both the M-GIS server and the Deegree WFS ran on the Apache Tomcat servlet container. The XSL Transformation is done through the Java API for XML Processing (JAXP)[3].

5   Technological Assessment

As described in previous sections, M-GIS revealed to be a stable and agreeable prototype, with a very interesting degree of interaction with the user and fulfilling the initial requirements of interoperability towards geographic information data-sources and of platform independence.

   The interpretation of the SVG document in the M-GIS client was built on top of kXML parser [Enh03], since available implementations of SVG renderers required MIDP 2.0 [Sun02] to run. This approach, allowed the prototype to execute on most graphical-enabled mobile platforms, because of the wide implementation of MIDP 1.0 [Sun00], but introduced some degree of inefficiency in the client application.

   Benchmarking demonstrated yet another bottleneck in M-GIS: the stylesheet-based transformation between GML and SVG. This transformation, implemented in this prototype through JAXP API, requires the creation of DOM trees, traditionally greedy in memory usage, that can turn out to be too lengthy if the requested area is too large or the number of features is too high.

6   Conclusions and Future Work

The main goal of the work described in this paper was to investigate the possibility of designing and implementing a mobile system, able to visualise and manipulate geographic information, with the interoperability of GIS systems in mind, i.e., by using formats and open standards. To assess this premises, an architecture was designed and a prototype called M-GIS was implemented.

   The results of the M-GIS assessment conducted allow us to conclude that it is possible, but with some limitations. The main limitations are related with the amount of information that can be processed by the mobile client. The first limitation of the current devices have to do with the amount of memory needed to visualise a given map and the second with the amount of time needed to process it.

   Another limitation of M-GIS is related with the graphical appearance, as the maps presented to the user are very simple due to the simplicity of the MIDP

---

[3] Due to space limitations, sections on Transforming GML to SVG and on SVG Interpretation can only be seen in the online version of this paper available at `http://usic.inescporto.pt/papers/mgis_egov04.pdf`.

graphics API used. This limitation could be overcome if vendor specific API were used but that would compromise the portability of the client application.

This limitation clearly indicate that M-GIS becomes more unusable as the amount of geographic information to be transformed and transmitted grows.

Besides the obvious improvements, like re-writing the GML to SVG transformation (server-side) and using native support for SVG rendering (client-side), the information layers can be annotated with maximum and minimum visible scale (for example, the edifications layer is only visible above the 1:5000 scale), reducing both transformation and transmission time.

In order to provide a better service, both for citizens and professional users, several new functionalities are thought and can be added to M-GIS:

Current location Show the map of the place were the user is at the time. This service could be targeted at people visiting the city and requires support in the client device for identification of the place (in cell phones it is enough to use the information of the current cell).

Points of interest Providing a list of interest points in the area being visualised could be added to help in finding, for example, the police or firemen stations in the area.

Global search At the moment, search is confined to the area being visualised. A global search could be implemented allowing finding a place (a Street for example) and changing the visualisation area to that place, i.e., getting a new map centred in the place.

References

[Car03]   Jorge Cardoso. M-GIS: Mobile Geographic Information System. Report, INESC Porto, 2003.

[Dee03]   Deegree. Deegree Project, 2003. `http://www.deegree.org`.

[Enh03]   Enhydra.org. Projecto kXML, 2003. `http://kxml.enhydra.org`.

[ESR03]   ESRI. ArcPad, 2003. `http://www.esri.com/software/arcpad/index.html`.

[Geo03]   Geo InSight. GeoGIS, 2003. `http://www.geoinsight.com`.

[OGC01] OGC. *Geography Markup Language (GML) 2.0*. Spec., OGC, Feb 2001.

[Poc03]   Pocket Systems Ltd. PocketGIS, 2003. `http://www.pocket.co.uk`.

[Sun00]   Sun Microsystems. Mobile Information Device Profile 1.0a. Spec., Dec 2000.

[Sun02]   Sun Microsystems. Mobile Information Device Profile 2.0. Spec., Nov 2002.

[W3C99] W3C. XSL Transformations, 1999. `http://www.w3.org/TR/xslt`.

[W3C03] W3C. Scalable Vector Graphics, 2003. `http://www.w3.org/Graphics/SVG`.