Centre for Geo-Information

Thesis Report GIRS - 2004 - 20

THE USE GML IN LOCATION-BASED SERVICES

A case study using a object-maintenance application

Rob Kromwijk

April 2004

WAGENINGEN UNIVERSITY
WAGENINGEN UR

THE USE GML IN LOCATION-BASED SERVICES

A case study using a object-maintenance application

Rob Kromwijk

Registration number: 48 00 50 291 179

Supervisors

Ron van Lammeren

A thesis submitted in partial fulfillment of the degree of Master of Science at Wageningen University and Research Centre, the Netherlands

April 2004
Wageningen, the Netherlands

Thesis code number: GRS - 7xxxx
Wageningen University and Research Centre
Laboratory of Geo-Information Science and Remote Sensing
Thesis Report GIRS - 2004 - 20

Preface

How time flies. After four years of study in Utrecht, Wageningen University became my next challenge. I can still recall the first time I visited Wageningen University, just two years ago. It was a warm day in the summer and mister Epema welcomed all the new students of the Msc course, geo-informationscience. Also I met my co-students for the coming two years, both Dutch and foreign students. I did not know what to expect, but working in an international group turned out to be the best learning experience of Wageningen University (together with the study itself). One can speak of a group feeling and co-students became friends.

Besides thanking of my co-students for the wonderful two years in Wageningen, I want to thank the professors, teachers, lectures and assistants, which were willing to show most ins and outs of the geo-information world. Without wronging anyone, I want to say special thanks to Ron and Aldo for their assistance and supervision during this research project, which I have been working on for the last seven months. Lastly and most of all, I want to thank my parents for their everlasting support and believe.

Thank you all

Contents

Abstract

The capabilities of mobile devices are ever increasing since the arrival of the first mobile handsets. Latest trends are Location-based Services; services provided to the user based on its location. These wireless developments did not stayed unnoticed in the field of Geographical Information Systems and several professional GIS applications for mobile devices are available. These services largely depend on geographical information, made available by many service providers, which contributes to a heterogynous geo-information market and thus data compatibility problems. The Open GIS Consortium has been brought into life in order to provide widely supported specifications for open interfaces and protocols. A breakthrough has been reached by the release of the Geography markup Language specification in 1999. GML is a universal format for structured documents and data, defined for the encoding, transport and storage of geographical information, including both geometry and properties of geographic features.

The combination of GML and Location-based Services seems to holds a great promise for the future of mobile spatial applications and services. In order to investigate both GML and LBS, a mobile object maintenance application has been developed with GML as its key aspect. The application is able to send the coordinates and corresponding attribute information of point objects to a server, where it immediately becomes available for other users, indoors or outdoors. The initial attempt to use GML as the exchange format did not succeeded, due to various limitations. However, the application does successfully make use of GML as a storage location for geographical data on the server, making it exchangeable with third parties, without any conversion or loss of data.

This report describes the main characteristics of Geography Markup Language and the way it is used on mobile devices at the moment. The implementation of the designed LBS is treated, together with an explanation of the choices being made. The application is tested and the report ends with conclusion and recommendations about the wireless application and the use of GML in the process.

Abbreviations

AGPS	Assisted Global Position System
ASCII	American Standard Code for Information Interchange
ASP	Active Server Page
CAD	Computer Aided Design
CPU	Central Processing Unit
CRS	Coordinate Reference System
CSS	Cascading Style Sheet
DIGEST	Digital blablablabla
DTD	Document Type Definition
DXF	Drawing exchange format
E-OTD	Enhanced Observed Time Difference
ESRI	Environmental Systems Research Institute
GI	Geographical information
GIS	Geographical Information System
GML	Geographical Markup Language
GPS	Global Positioning System
GPL	General Public License
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hyper Text Transfer Protocol
IIS	Internet Information Service
ISO	International Organization for Standardization
JPEG	Joint Photographic Expert Group
LBS	Location-based Service
MBR	Minimum Bounding Rectangle
MMS	Multimedia Message Service
NEN	Nederlands Normalisatie instituut
ODBC	Open Database Connectivity
OGC	Open GIS Consortium
OMA	Open Mobile Alliance
PDA	Personal Digital Assistant
PHP	PHP: Hypertext Preprocessor

RDBMS	Relational Database Management System
RDF	Resource Description Framework
SDTS	Spatial Data Transfer Standard
SGML	Standard Generalized Markup Language
SMS	Short Message Service
SRID	Spatial Reference Identifiers
SRS	Spatial Reference System
SQL	Structured Query Language
SVG	Scalable Vector Graphics
TOA	Time of Arrival
TDN	Topografische Dienst Nederland
UOM	Unit of Measure
URL	Uniform Resource Locator
UTMS	Universal Mobile Telecommunications System
W3C	World Wide Web Consortium
WAP	Wireless Application Protocol
WBMP	Wireless Bitmaps
WFS	Web Feature Server
WML	Wireless Markup Language
WMS	Web Map Server
WWW	World Wide Web
XML	eXtensible Markup Language
XSD	eXtensible Markup Language Schema Document
XSL	eXtensible Markup Language Style sheet Language
XSLT	eXtensible Markup Language Style sheet Language Transformation

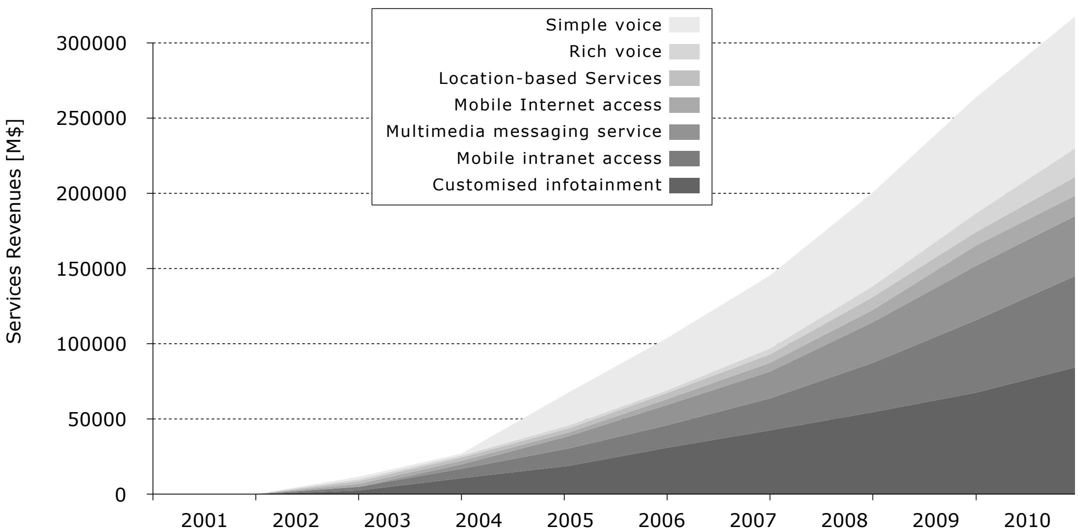
1 - Introduction

1.1 – Background

The ones that have not been asleep for the last decade have noticed a high increase of mobile services, starting with cellular phones, enabling voice transfer, but today's devices are decreasing in size and increasing in capabilities. SMS, MMS, I-mode and WAP are well-known terms these days in the field of mobile communication and LBS can be added to that list. Location-based Services are described as:
'The capability to find the geographical location of a mobile device and provide them services based on the location information' [1, 2]

LBS are characterized by their ability to support itinerant (while moving with a user), distributed (integrating functions that are performed at different place, transparent for a user) and ubiquitous (delivering the same functionality independent of the user's location) computing. *[3]*

Figure 1-1:
Graph showing the world-wide revenues by mobile services [UTMS forum, 2001].



The various Location-based services can be classified by their functionality and utilization of location information. 'Pull' services are submitted by the user of the mobile device, like a request for their current position. However, often only coordinates are not sufficient; so a map indicating the user's position would be more helpful. This service can be extended to a guidance service by providing access to points-of-interest or even a yellow paging service by providing additional information. In this way, LBS are an evolving process. Functional services are different type of LBS, which differ from information services by supplying functions instead of information, e.g. ordering a taxi. Besides 'pull' services, location-aware or 'push' services exist, which are very attractive for commercial businesses. Service provider will send a push service when a user is within proximity of an object, like sending advertisements. These Location-based Services make use of the mobile user's dynamic location and some additional static information. LBS can even be extended by adding a second dynamic component, like a moving object and even tracking or finding multiple objects is possible. *[4]*

1.1.1 – Location-based Services and GIS

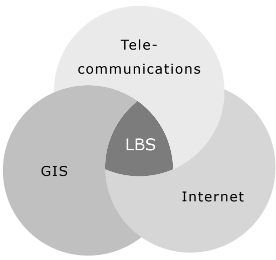


Figure 1-2: Location-based Services have emerged from communication, Internet and GIS businesses.

Last few years' users in the field geo-information are focusing more on Location-based Services. Not surprisingly, since LBS evolve around geographical information. Some even say that LBS are like wireless GIS, but LBS and GIS have different roots. Geographical Information Systems have been developed on the basis of geographical data applications. LBS's are evolved out of public mobile services. Many pilot projects have been launched to investigate the possibilities, advantages and usefulness for the use of mobile devices in GI-related businesses. Many commercial GIS vendors see the potential of wireless geo-services and are developing mobile GIS applications. However, research institutions make the best results and a real commercial mobile breakthrough have not yet occurred. Mostly a lack of standards and integration cause this delay. Interoperability determines the success of applications and services in the long run. Seamless integration of all components into a reliable Location-based Service requires open standards for data exchange and application interfaces. Conventional methods have distinct disadvantages in terms of interoperability and integration and do not allow much flexibility in cost and scalability. *[5]*

Although interoperability can be defined at different levels, in the context of geo-information, interoperability concerns Geographic Information System or Services. GIS needs special requirements concerning interoperability, because of the complexity of geo-information together with difficulties in institutional and legal issues. Interoperability in a GI-environment is often defined as:
'*Geographic interoperability is the ability of information systems to freely exchange all kinds of spatial information about the Earth and about the objects and phenomena on, above, and below the Earth's surface; and secondly co-operatively, over networks, run software capable of manipulating such information'. [6]*

Often a misperception is made regarding this definition. The definition does not assume that every data provider adopts the same data format; rather it addresses the ability to understand each others format.

However, many vendors often result in many data formats and compatibility problems. Every GI-vendor has its own philosophy about how to store and use geographical data. Spatial Data Transfer Standard (SDTS) and AutoCAD's drawing exchange format (DXF) are examples of data exchange formats, but often they are de-facto standards, so still no real vendor independency. Throughout the years interest groups are established in order to investigate the use of standards in the geo-information sector, like the International Organization for Standardization (ISO), Open Mobile Alliance (OMA) and the Open GIS Consortium (OGC). Especially the OGC is focused into prescribing GI-standards in order to promote interfaces and protocols that enable interoperable geo-processing.

1.1.2 – Open GIS Consortium

The Open GIS Consortium is an international industry consortium of more than 230 companies, government agencies and universities participating in a consensus process to develop publicly available geo-processing specifications. Open interfaces and protocols defined by OGC specifications support interoperable solutions that 'geo-enable' the Web, wireless and location-based services. *[7, 8]*
The OGC divides the term interoperability regarding geo-information into seven parts.

- GI (spatial data) discovery
- GI access
- GI integration from multiple sources
- GI display in a single view
- GI analysis
- GI display generation when multiple sources and types of data are involved
- GI based enterprise information system assembly with features from many software providers *[6]*

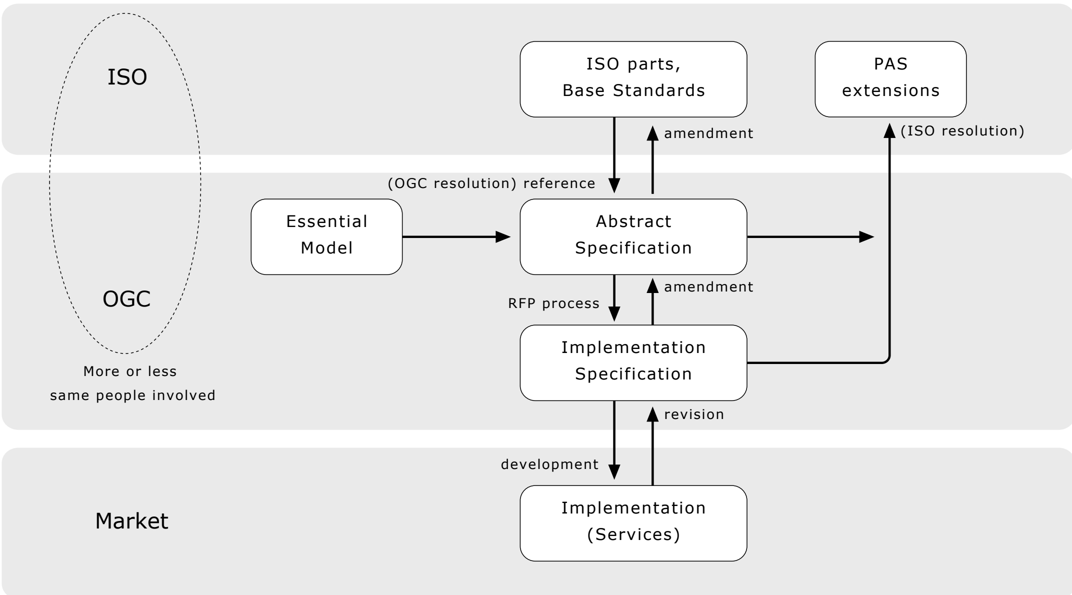


Figure 1-3:
The interoperability model from the Open GIS Consortium, which guides the development of OGC specifications.

The interoperability program of the Open GIS Consortium consists of development initiatives, infrastructure initiatives and OGC specifications. Open standard efforts are developed and tested with the use of testbeds. The forthcoming draft specifications are reviewed, revised and approved in the formal Open GIS Specification Adoption Process. An important specification, which has been approved recently, is the Geography Markup Language Implementation Specification. GML is an eXtensible Markup Language (XML) based data interchange format, evolved as a clear choice for use with geo-information. GML uses XML encoding for the modeling, transport and storage of geographic information including both the spatial and non-spatial properties of geographic features. *[9]* It provides a variety of objects for describing geography, including features, coordinates, geometry, topology and time. Geography Markup Language defines the XML Schema syntax and conventions that:

- Provide an open framework for the definition of geo-spatial application schemas and objects
- Allow profiles that support proper subsets of GML framework descriptive capabilities
- Support the description of geo-spatial application schemas for specialized domains
- Enable the creation and maintenance of linked geographic application schemas and datasets
- Support the storage and transport of application schemas and data sets
- Increase the ability of organizations to share geographic application schemas and information

The development and promotion of geo-information, in particular the use in combination with Location-based Services, depends largely on the use of open standards. Therefore it is important to investigate the concept of GML for storage and exchange of geographical information, because the future of GIS integration with other services depends on it.

1.2 - Problem Definition

The use of open standards becomes more important by the day. The ability of using data format standards has opened the eyes of many GIS vendors. Especially within Location-based Services, which are more utilized outside the field geo-information than inside, data exchange standards are a must. They require open standards that allow data content to be easily transferred and displayed across wireless applications on a variety of mobile devices. *[10]* Although the Open GIS Consortium recommends Geographical Markup Language is a clear choice for exchanging geo-information, in practice not every GI-vendor is using it, although most vendors are adapting GML into their software solutions.

When it comes to wireless devices, GML possibilities are even more limited; mainly due to hard- and software limitations of mobile devices. Looking at smartphones, the main drawbacks are processing speed, storage capacity and bandwidth for wireless data transfer. However, its physical size gives a high potential for use in the field. Still many geo-information employees, who are working outdoors, are using pen and paper to make notes about their findings and their findings have to be entered in the computer at the office; very labor intensive and superfluous with the arrival of mobile devices.

Location-based Services have the potential to become one of the most important fields within the GIS and will move GIS employees outdoors, especially when hardware drawbacks are no longer an issue. Although hardware components are important for the success of LBS, also the contribution of data exchange format cannot be ignored.

1.3 - Research Objective

Because data formats is often underexposed in many LBS studies, this research project will focus on the use of GML in Location-based Services, since GML is advocated as the open solution of data exchange problems. An application can be designed, which includes the issue of GML in Location-based Services. A suitable type of LBS, which fulfill the needs of GI-workers, is maintenance of topographic objects in the field. When a user is in the field, he has to receive data via a wireless connection, based on its location (via GPS). After receiving the requested data, objects can be added, removed or updated and send back to the central data storage location. This type of application can also be used for any kind of inventory purposes. To improve interoperability and the use of open standards, GML is a necessity. Furthermore, it gives the possibility to use spatial data for other purposes and third-party software.

To summarize the research objective in one sentence:
'To investigate and give recommendations about the use of Geographic Markup Language in Location-based Services by incorporating it as an integrated part of a maintenance application'.

This research objective can be divided into three main topics: 'The concepts of Location-based Services', 'the application for updating topographic objects in the field' and 'the use of XML/GML (in LBS)'. Within each topic questions have to be answered, which will be lead to the fulfillment of the research objective. The main research questions are:
- What are the main characteristics of XML/GML and how is it used in today's applications?
- How should the desired Location-based Service look like and how are the components implemented in the application?
- How is GML embedded in the application and what are its advantages for wireless applications?

1.4 – Report outline

In order to fulfill the research objective, a step-to-step approach of the activities have to be performed, which is given in figure 1-4. These five steps can be found back in the various chapters of this report (the numbers represent the various chapters).

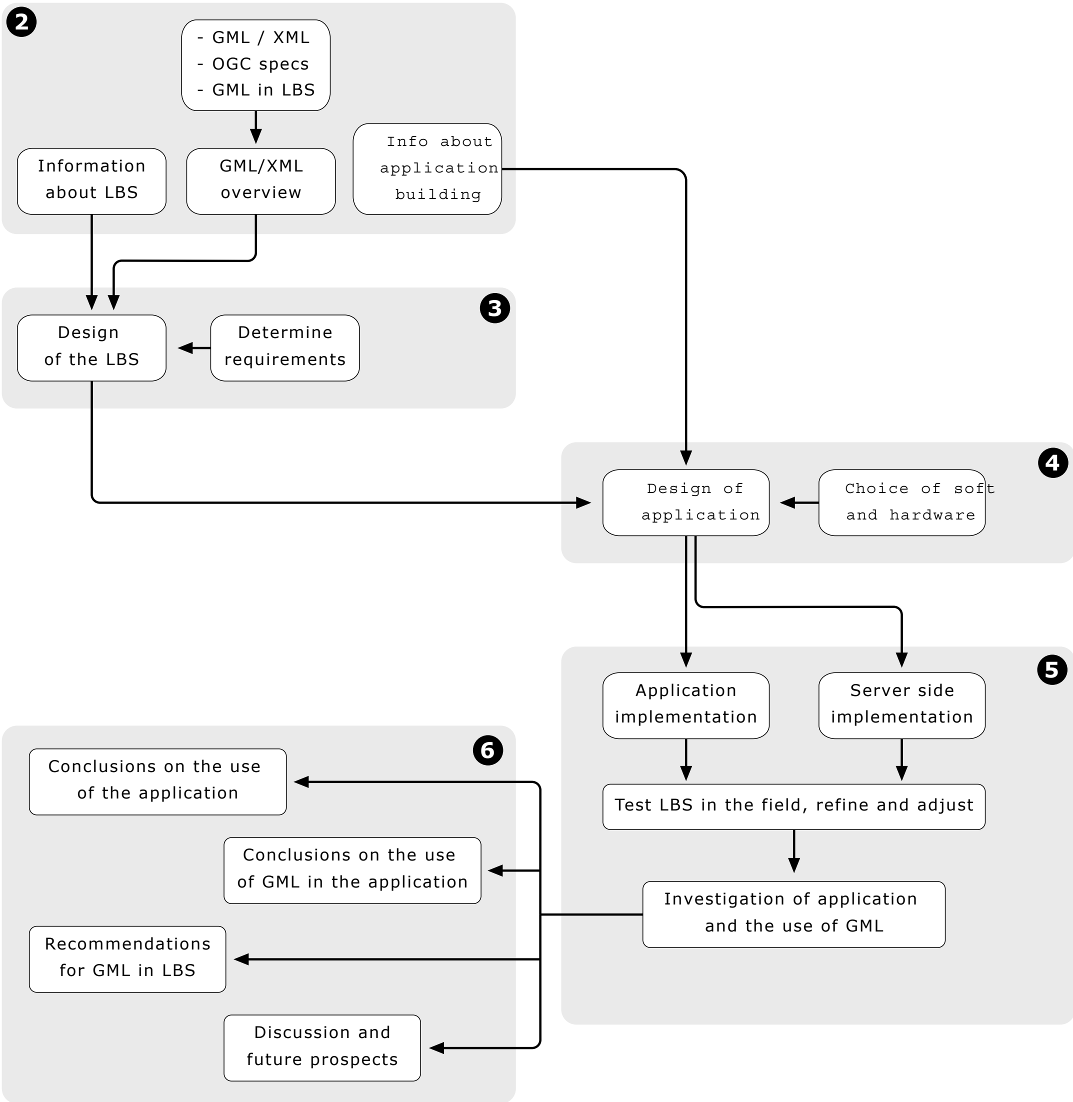
Chapter two deals with the investigation of XML, GML and LBS concepts. In order to develop an application, suitable for supplying Location-based Services, profound insight has to be gained into LBS in the field of geo-information. Literature about open standards and their use in LBS has to be studied and experience with both markup languages has to be gained. During the research project additional work has been done for the 'Topografische Dienst Nederland', which is about the restructuring of the data sets to the new GML standard. In this chapter some attention will be given to the TOP10NL project.

Chapter three is about the design of the proposed LBS. With the gathered knowledge, an inventory will be made of the various components of the application and the user requirements. Chapter four deals with the translation from the LBS design to the components of the application, supplying the service. Choices of hard- and software are explained in order to construct the proposed LBS. Chapter five explains to application in detail at user level. Furthermore, the application is being tested and findings are given in this chapter.

Chapter six deals with the interpretation of the findings, described in chapter five. It evaluates the usefulness of the application and conclusions are given about the exchange via GML, the application and the combination of GML and LBS. The report will end with a discussion and recommendations of the project, application and GML.

Figure 1-4:

The flowchart shows the various steps that are taken in the research project. The numbers represent the different chapters of this report.



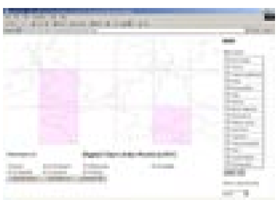
2 – Standardization

Figure 2-1:
Four examples of online Web Map Server functionalities.

www.moxiemedia.com/imf_sites.html

www.demo.degree.org:8080/client/control?action=init

www.svgmapper.com/example/example2/land-register.htm



www.web-maps.com/geotech/svg-dcw/world-fr.htm

In an ideal situation every Geographical Information System is independent of platform, device or application and allowing for sharing of geographic data, integration among different GIS technologies, and integration with other non-GIS applications. *[11]* New developed solutions should not be tied to one single vendor. In fact, the geographical data challenge is substantial as long various geo-data products, in different formats and different languages co-exist. In the geographical information world, key players have been united in the Open GIS Consortium (OGC). It is a non-profit organization founded in 1994 to address interoperability among systems that deal with geographical data. By means of its consensus building and technology development activities, OGC has a significant impact on the geo-data environment and geo-processing standards. *[12]* The OGC is not the only organization, dealing with standardization; TC 211 is part of the International Standards Organization (ISO) and was formed for the development of standardization in the GI-field.

Organizations and companies have been providing online mapping services for years. However, as a result of this isolated development, online mapping services from different vendors cannot interoperate. To address this problem, the Open GIS Consortium developed a non-proprietary web mapping approach based on open interfaces, encoding and schemas. The Specification and Interoperability Program provide an industry consensus process to plan, develop, review and officially adopt Open GIS Specifications for interfaces, encoding and schemas that enable interoperable geo-processing services, data, and applications. Hundreds of billions of dollars worth of digital maps and earth images, which until now could not be accessed and used without special skills and software, can now become an integrated part of the broader information infrastructure.

There are different types of interoperability. Interoperability at data level requires involvement in the development of standards for data descriptions (catalogues and reference data), data access (database interfaces) and data transport (representation and protocols). The basic idea is that shared data is stored only once and maintained by the producer. In this way, data is up to date and no redundant versions need to be stored. Institutional interoperability at data level requires involvement in the development of standardized workflows and standards for security, data protection and electronic commerce; accompanied by a legal framework on ownership, copyright, privacy, and by transparent public information policies.

Thus, the need for standards for the geographical data exchange is evident and accounts for both wireless and none-wireless services. Furthermore, seamless integration into a reliable Location-based Service (LBS) requires open standards for data exchange and application interfaces. *[5]*

The eXtensible Markup Language seems to be the solution to the data exchange issue, although conflicting sounds can be heard. Some like the rich structure and semantics, while others complain about excessive complexity. *[13]* In order to give answers on the 'exchange via XML' issue, insight has to be gained on the concept of XML, and especially geo-XML, also known as Geographical Markup Language (GML). In this chapter an overview will be given about the XML/GML technology, followed by a closer look on GML use in Location-based Services.

2.1 - eXtensible Markup Language

eXtensible Markup Language (XML) is a universal format for structured documents and data. It is designed and developed as a way to share data via multiple systems, platforms and applications by strictly defining how data has to be stored and accessed. It is license-free, vendor/platform-independent and is supported by a growing number of individuals and organizations. XML is a strict simplified subset of the Standard Generalized Markup Language (SGML) and is a W3C standard since 1989. It allows generic SGML to be served, received, and processed on the Web in a manner similar to what is done with HTML today. *[14]*

However, XML goes beyond the Hyper Text Markup Language. The main difference between XML and HTML is the extensibility; new features cannot be added to HTML, prohibited by the language definition. Furthermore, XML is a meta-language, used to define other languages (like HTML) and tag names are used to describe the content and it accommodates the meaning of the data contained. HTML also makes use of tags, but most tags are designed only for presentation purposes. *[15, 16]*

The main reason why XML is so widely introduced are its advantages. First of all, the XML syntax is license-free and platform independent. Secondly, no ambiguity can occur in XML interpretation, because XML is text-based and the specification is explicitly based on other international standards for the exchange of text. It can be read by simple text editors and is even human readable. Furthermore, the language is flexible and extensible, making it suitable for different purposes and users. *[17, 18]*

Actually, XML documents can consist of three different parts: content, structure and presentation.

2.1.1 - XML content

In order to explain how a XML document is build up, a simple example is given first.

Fragment 2-1: example of how content of an eXtensible Markup Langauge document (extension: xml) is built-up.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemalocation="http://
www.wur.nl/schema.xsd">
<building id=10201>
 <name>Alterra</name>
 <address>
  <street>Droevedaalsesteeg</street>
  <number>3</number>
  <city>Wageningen</city>
 <function>research center</function>
</building>
```

A XML document starts with a heading, containing information about the document and the location of the XML structure document. The basic unit of an XML document is an element; ‘building’ is an element and ‘name’, ‘address’ and ‘function’ are its sub-elements. Each element has a name and has content between a start tag (<>) and an end tag (</>). A sub-element can contain other sub-elements, like ‘address’ and its sub-elements ‘street’, ‘number’ and ‘city’. Besides a name, an element can contain one or more attributes, directly after the name (‘id’ is an attribute of ‘building’). Be aware that an attribute in XML is not the same as attribute in database terminology. A database attribute can be imported into XML as an XML-style attribute, but often it will be implemented as a child element of the object element. *[19, 20]*

2.1.2 - XML structure

XML documents are not fixed to pre-defined set of tags. As a result the type and order of the elements are not a fixed set. Users can create own elements, which result in various document structures. Thus an instance XML document (like fragment 2-1) cannot directly be read by an application. First, the application must know which elements can occur and their order. XML structure documents contain or point to markup declarations that provide a grammar for a class of documents. *[21]* This implicates that a structure document can be used for many content documents, as long as structure correspond. The structure of an XML document is described in a Schema document or a Document Type Definition (DTD). They both have the same purpose; however, a Schema document has XML as its technical format, meaning that it can be processed with the same tools as an XML document. Furthermore, a Schema is more flexible and has more possibilities to incorporate constraints in the data model.

When a program validates a XML document against its Schema, it checks that the element types and attributes are as defined in the Schema. The declarations are meant to proclaim the names of the elements that can appear in the XML document, while the type definitions are used to specify the nesting of elements and the sequence in which they will appear, although this process is not obligatory. However, the rules of well-formedness have to be obeyed; elements have to begin and end with tags and be correctly nested. *[17]*

Fragment 2-2: example of how the structure of an eXtensible Markup Langauge Schema document (extension: xsd) is built-up. This example corresponds with the content of fragment 2-1.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:annotation>example of XML Schema</xsd:annotation>
 <element name="building">
 <xs:attribute name="id" type="string/>
  <xs:complextype>
   <xs:sequence>
    <element name="name" type="xs:string"/>
     <element name="address" type="xs:string"/>
      <xs:complextype>
       <xs:sequence>
        <element name="streetname" type="xs:string"/>
        <element name="number" type="xs:integer"/>
        <element name="city" type="xs:string"/>
       </xs:sequence>
      </xs:complextype>
     <element name="function" type="string"/>
    </xs:sequence>
   </xs:complextype>
  </element>
</xs:schema>
```

This XML Schema corresponds with the XML Document of fragment 2-1. ‘Building’ has an attribute called ‘id’ and consists of a sequence of sub-elements. There is a basic difference between complex types, which allow elements in their content and may carry attributes and simple types. Furthermore, there is a distinction between definitions, which create new types and declarations, which enable elements and attributes with specific names and types to appear in document instances.

Each element in the Schema has a prefix ‘xs:’, which is associated with a XML Schema namespace through the declaration ‘xmlns:xs=‘http://www.w3.org/2001/XMLSchema’’. Although XML is a universal format problems can occur. Firstly, it is possible that multiple XML Schemas use the same term to express different concepts, like ‘name’. As long as document instances are used within the same domain, the elements will be interpreted correctly. Also, the opposite problem occurs when different names are used for the same concept, like ‘author’ and ‘creator’.

The W3C ‘Namespaces in XML’ specification addresses this problem. It allows names to be qualified by associating them with a namespace. A namespace is a collection of names and the uniqueness of namespace names is guaranteed by making them URL’s. Element type names and attribute names can then be qualified by a namespace name, which removes the ambiguity where two names from different schemas collide. At the syntactic level, the association between the local part of the name and the namespace name is made through the use of a namespace prefix. The prefix is associated with the namespace name (by a namespace declaration), and the prefix is then used to qualify names. *[17]*

2.1.3 - XML presentation

Because XML is not a fixed set of tags, a browser does not know in advance which tags it will encounter in a XML document and how to render each element. The browser needs explicit instructions how to render each XML tag, which re supplied by a third document. *[22]* One of the possibilities is the use of a Cascading Style Sheet (CSS). Nowadays CSS is mostly used in association with HTML files, but CSS can be applied to XML documents as well. Within a CSS, all used XML tags are declared and the way they should be rendered. In the XML document a reference is made to the Style sheet.

A more used solution is a XML Style sheet. It defines how the tags and content of the tags should be visualized, according to the creator of the document. *[23]* In fact, a XSL document uses the same XML tags, but with instructions for visualization. An XSL style sheet processor takes care of the input XML document or data in XML and produces the presentation of that XML source content, also known as XSL Transformation (XSLT).

An example matching fragment 2-1 is given:

Fragment 2-3: example of how an eXtensible Markup Langauge Style Sheet document (extension: xsl) is built-up. This example corresponds with the content of fragment 2-1.

```
<?xml version=”1.0” encoding=”iso-8859-1” ?>
<xsl:stylesheet version=“1.0” xmlns:xsl=“http://www.w3.org/1999/XSL/Transform” xmlns:
xs=“http://www.w3.org/2001/XMLSchema”>
<xsl:template match=“/”>
 <html>
  <head />
   <body>
    <xsl:for-each select=“Building”>
     <xsl:for-each select=“@id”>
      <span style=“font-weight:bold; “></span>
      </xsl:for-each><br />
      <table border=“1” width=“100%”>
```

```
      <tbody>
       <tr>
              <td width="10%">naam:</td>
               <td>
                <xsl:for-each select="name">
                 <xsl:apply-templates />
                  </xsl:for-each>
               </td>
       </tr>
       <tr>
              <td width="10%">straat:</td>
               <td>
                <xsl:for-each select="street">
                 <xsl:apply-templates />
                  </xsl:for-each>
               </td>
       </tr>
       <tr>
              <td width="10%">nummer:</td>
               <td>
                <xsl:for-each select="number">
                 <xsl:apply-templates />
                  </xsl:for-each>
               </td>
       </tr>
       <tr>
              <td width="10%">plaats:</td>
               <td>
                <xsl:for-each select="city">
                 <xsl:apply-templates />
                  </xsl:for-each>
               </td>
       </tr>
       <tr>
              <td width="10%">functie:</td>
               <td>
                <xsl:for-each select="function">
                 <xsl:apply-templates />
                  </xsl:for-each>
               </td>
       </tr>
      </tbody>
     </table>
    </xsl:for-each>
   </body>
  </html>
 </xsl:template>
</xsl:stylesheet>
```

A browser display of fragment 1-1 will look as follows (and repeated for every building in the list):

Table 2-1: Browser display example of the XML content of fragment 2-1, displayed using XSL of fragment 2-3.

10201

Naam:	Alterra
Straat:	Droevendaalsesteeg
Nummer:	3
Plaats:	Wageningen
Functie:	Research center

It becomes clear that XML Style sheet Language makes use of ‘search and replace’ techniques with special keywords for constructing XML output. The main difference between XSLT and CSS is the complexity. XSLT is more complex and for one single XML document CSS is best to use. However, XML and XSLT are most useful for publishing multiple documents to multiple audiences from a single set of XML source files.

2.2 - Geography Markup Language

In previous paragraphs, the concepts of XML documents and Schemas were explained. Geography Markup Language (GML) is a special kind of XML Schema (sometimes referred to as geo-XML), defined for the encoding, transport and storage of geographic information, including both geometry and feature properties. GML is based on the abstract model of geography, designed by the Open GIS Consortium, which describes the world in terms of geographic entities or features. A geographic feature is *'an abstraction of a real world phenomenon; it is a geographic feature if it is associated with a location relative to the Earth'. [9, 15]*

In 1999, the OGC reached agreement on the GML specifications. The second version had multiple improvements, like the support of XML Schema (XSD). XML Schema has the ability to support type inheritance, distributed schema integration and namespaces. *[24]* Furthermore, this version is concerned with simple features: features, whose geometric properties are restricted to 'simple' geometries whose coordinates, are defined in two dimensions and the delineation of a curve is subject to linear interpolation, like points and lines. *[9]* GML does not support 3D geometry constructs directly, although 3D coordinates are allowed. GML can take advantage of the XLink and XPointer mechanisms enabling linking of features in one GML file to those in another XML file through hyperlinks. As a result, a new, even virtual, dataset can be constructed by setting up links between two or more primary GML data stores.

At this moment, version three of the specification has been released, again with several improvements:
- Representing geo-spatial phenomena in addition to simple 2D linear features, including features with complex, non-linear, 3D geometry, features with 2D topology, coverages and observations
- Providing more explicit support for features' properties and other objects whose value is complex
- Representing spatial and temporal reference systems, units of measure and standards information
- Use reference system, units and standards information in the representation of geo-spatial phenomena, observations, and values
- Represent default styles for feature and coverage visualization

2.2.1 - GML content

Just like XML, Geographic Markup Language can be divided into content, structure and visualization. The syntax is similar and tags are used to divide spatial and non-spatial elements. *[23]* In the next fragment, an example is given of an object, described in GML, called road. It is a line feature, but it is also defined as an edge for topology purposes. The non-spatial part is followed by the spatial component of the feature.

Fragment 2-4: example of how content of a Geography Markup Langauge document (extension: gml) is built-up. GML elements can be recognized by the gml prefix.

```
<?xml version=”1.0” encoding=”iso-8859-1” ?>
<FeatureCollection xmlns:gml=“http://www.opengis.net/gml” xmlns:xsi=“http://www.w3.org/2001/
XMLSchema-instance” xsi:schemaLocation=“//imsgrs.wur.nl/gml_schema.xsd”>
<gml:boundedBy>
 <gml:Box srsName=“EPSG:4326”>
  <gml:coordinates>43.0536232850659,-89.4696159042373 43.0637170397021,-89.4575395800512</
gml:coordinates>
 </gml:Box>
</gml:boundedBy>
<gml:FeatureMember>
 <road>
  <road_num>70919412018</road_num>
```

```
   <name>Dorpstraat</name>
   <ID>17363</ID>
<topology>
 <gml:edge gml: id="e1">
   <gml:directedNode orientation="-" xlink:href="#n1"/>
    <gml:directedNode orientation="+" xlink:href="#n2"/>
     <gml:centerLineOf>
      <gml:LineString>
       <gml:coordinates>10,10 20,10</gml:coordinates>
      </gml:LineString>
     </gml:centerLineOf>
    </gml:Edge>
</topology>
 </road>
</gml:featureMember>
</gml:FeatureCollection>
```

2.2.2 - GML structure

GML uses an explicit syntax, referring to the Feature model defined in the OGC Abstract Specification. Both features and properties of features are encoded as XML elements, whose name is the name of the property. The value of a property may be simple or it may be a feature or other complex objects and can be encoded inline or set as a reference via Xlink The term property is used to refer to a GML property, which is any characteristic of a GML object and is a GML property, if only it is a child element of a GML object element. The meaning of each property is indicated by the name of the element that instantiates it. The result is a layered XML document, in which XML elements correspond to features, objects or values. The function of a feature, object or value can be determined by looking at the name of the property element, which directly contains it or which carries the reference to it.

Some basic components for constructing identifiable GML objects are described in the following Schema example, which corresponds with fragment 2-4.

Fragment 2-5: example of how the structure of a Geography Markup Langauge Schema document (extension: xsd) is built-up. This example corresponds with the content of fragment 2-4. GML elements can be recognized by the gml prefix.

```
<?xml version="1.0"  encoding="iso-8859-1" ?>
<xsd:Schema xmlns:gml="http://www.opengis.net/gml" xmlns:xsd="http://www.w3.org/2001/
XMLSchema" elementFormDefault="qualified" version="1.0">
<xsd:import namespace="http://www.opengis.net/gml" schemaLocation="feature.xsd" />
<xsd:element name="featureCollection" type="gml:featureCollectionType"
substitutionGroup="gml:_FeatureCollection" />
<xsd:complexType name="featureCollectionType">
</xsd:complexType>
<xsd:element name="featureMember" type="gml:featureMemberType" substitutionGroup="gml:
_FeatureMember" />
<xsd:complexType name="featureMemberType">
<xsd:sequence>
<xsd:element name="parcel_num" type="integer" minOccurs="0" />
<xsd:element name="owner_name" type="string" minOccurs="0" />
<xsd:element name="ID" type="integer" minOccurs="0" />
<xsd:element name="Topology" type="gml:EdgeType/>
<xsd:element name="Geometry" type="gml:LinePropertyType" minOccurs="0" />
</xsd:sequence>
</xsd:complexType>
</xsd:extension>
</complexType>
</xsd:Schema>
```

In fragment 2-5, only a few GML elements are used (which can be recognized by the gml prefix). There are various standardized XML Schema's developed for representing geographic data, each containing predefined GML elements. These schema documents alone do not provide a schema suitable for constraining data instances; rather, they provide base types and structures, which may be used by an application schema. GML schema documents are XML Schemas that define XML types and XML elements to encode GML objects, elements to encode GML properties of those objects and XML attributes qualifying those properties. An application schema declares the actual feature types and property types of interest for a particular domain, using components from GML in standard ways. *[25]* Pre-defined GML schema's effectively provide a meta-schema, from which an application schema can be constructed. The available predefined XML Schema's for encoding GML elements are given below (consult the OGC GML Specification *[9]* for an extensive list and explanation of predefined Schema's).

- Feature
  The Feature Schema provides a framework for the creation of GML features and feature collections. When using the Feature Schema from GML, it is necessary to identify what plays the role of features and what plays the role of properties. A GML feature has a set of properties, where the specific set of properties defines the feature type.

- Geometry
  The geometry model distinguishes geometric primitives, aggregates and complexes and is based upon the OGC Simple Features Model (figure 4-3). In addition there are coordinates, elements for encoding coordinates and a box element for defining the spatial extent. The Geometry Schema includes type definitions for abstract geometry elements, concrete point, line and polygon geometry elements as well as complex type definitions for the underlying geometry types and the geometry schema is by far the most complicated.

- Coordinate reference system
  GML requires a coordinate reference system (CRS) to be referenced whenever location coordinate information is given. The CRS instance will give information about the coordinates, including the order of the coordinates, the unit of measure that goes with each coordinate and the physical meaning in terms of its attachment to the earth. Also, it is sometimes necessary to express a conversion or transformation to another CRS and the definitions of coordinate conversions and transformations are also encoded using the CRS schemas.

- Topology
  Topology describes the properties of objects, which are invariant under continuous deformation. Most importantly, topology is used to accelerate computational geometry. The construct of topology allows definition of spatial relationships between objects using simple combinatorial or algebraic algorithms. Furthermore, it allows a compact and unambiguous mechanism for expressing shared geometry among geographic features and can consist of a solid, face, edge or node.

- Temporal information
  GML temporal schema extends the core elements of GML to include elements for describing the temporal characteristics of geographic data. Their purpose is to describe and provide a means of describing the history of a dynamic feature. The temporal schemas consist of two interrelated schemas: the main schema provides basic elements such as primitive types for representing temporal instants and periods and a more specialized schema defines types used to represent dynamic features.

- Definitions and dictionaries
  Many applications require definitions of terms, which are used within instance documents as the values of certain properties or as reference information to tie properties to standard information

values (like units of measure, descriptions). Components to support this functionality are provided in GML as a generic definition, which may serve as the base for more specialized definitions and a generic dictionary, which allows a set of definitions or references to definitions.

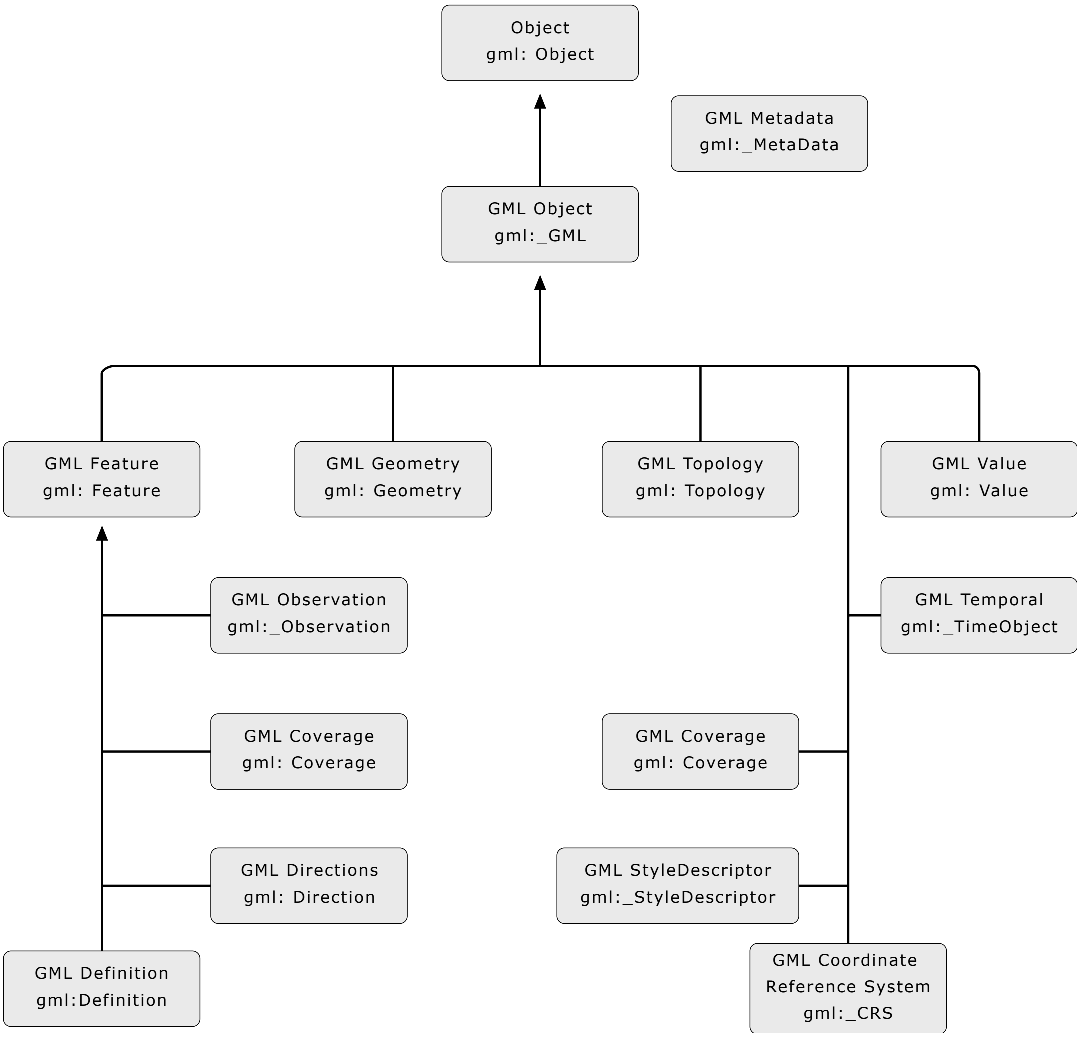


Figure 2-2: GML defines the various entities such as features, geometries, topologies etc. through a hierarchy of GML objects.

- Units, measures and values
  Several GML schemas introduce components that concern or require quantitative values, which use a reference scale or units of measure. The attribute 'uom' holds a URL, which refers to a resource that defines the units.

- Directions
  The direction schema provides the GML Application Schema developer with a standard property element to describe direction and associated objects that can be used to express orientation, direction, heading, bearing or other directional aspects of geographic features.

- Observation
  A GML observation models the act of observing, often with a camera, a person or some form of instrument. A GML observation is considered to be a GML feature with a time at which the observation took place and with a value for the observation. Observation can differ between tourist photos, images acquired by space borne sensors or a temperature measurement.

- Coverages
  Coverages support mapping from a spatio-temporal domain to attribute values, where attribute types are common to all geographic positions within the spatio-temporal domain, consisting of a collection of direct positions in a coordinate space. Examples of coverages include raster images, triangulated irregular networks (TIN), point coverages and polygon coverages. The information describing a coverage is conventionally represented: as a set of discrete location-value pairs or as a description of the spatio-temporal domain (e.g. grid).

- Default styling
  One of the requirements in developing GML is strict separation of content and presentation. Therefore none of the GML data documents has built-in styling information. However, a default styling mechanism is created to act as a plug-in to a GML dataset. The default style schema is directly related to the rest of the GML schemas only in that it imports some very basic constructs from it. The style information that is assigned to a data set may be used for styling, but may also be completely ignored. Since GML is feature-based encoding, GML default style always applies to a feature, features or feature collections.

The base schemas for GML described above have been modularized so that application schemas that do not need the complete set of GML definitions, but may import only subsets that are required. An application schema that needs definitions from more than one of these GML subset schema hierarchies can import gml.xsd and get all of the GML definitions. Or it can contain multiple imports for just the appropriate GML schema documents, thereby excluding unwanted GML type definitions.

Figure 2-3:
Scalable Vector Graphics is a method for displaying GML data, providing the advantage of scalability.

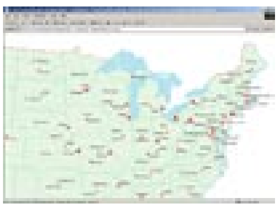
Moreover, when an application schema will be used in a processing environment that lacks CPU, memory or bandwidth, like mobile hand-held devices, an absolutely minimal import of GML definitions is often desired. The custom GML schema approach might bring in a large number of unwanted definitions from each GML schema. The solution is to create a single GML subset schema that contains exactly the required GML type and element definitions. However, creating such a GML subset schema by hand using a text or XML editor to cut and paste definitions is a tedious and error-prone process because it involves analyzing type definition dependencies across the many GML schema documents. *[9, 15, 24]*

SVG can display any vector-related data.

2.2.3 - GML visualization

SVG is suitable to display (GML) maps.

One of strengths of separating content and presentation within GML is the ability to represent the world independent of any particular visualization. By removing the limitations of one geographic representation, it becomes very suitable for geographic data transfer. GML must be styled for presentation. A presentation can be styled as a graphical form such as a map, but also text or even a sequence of voice instructions. *[15, 19]*



Details can be shown without pixelation.

There are several ways of styling GML features. First of all, the creator can supply a default styling, but it will only acts as a plug-in; thus the user can ignore this styling (see paragraph 2.2.2). Furthermore, GML files can be imported into a GIS or CAD application, where color and style can be added, using the default tools. When GML is not imported into another application, more possibilities are available. The GML can be accompanied with style information contained in a XML file, Cascading Style sheet or parameter file. Software reads both files and generates the model. Secondly, the application has an interactive module, which enables users to choose there own graphical styling. Also, combinations of both can occur.

GML data can be used for many different purposes on many different devices, which is an advantage, especially for Location-based Services. To make different maps from GML, 'map stylers' can be used to style GML elements into forms, which can be interpreted for graphical display in a browser, using graphical formats (e.g. Scalable Vector Graphics), but non-graphical presentation of the same data is possible. *[26]* SVG has the advantage of vector display and allows for three types of graphic objects: vector graphic shapes, images and text. The vector geometry types, which are defined within SVG, are rectangle, circle, ellipse, line, polyline, polygon, symbol and path. To each object type different style attributes can be defined. Furthermore, the integration of raster overlays allows the use of raster images in combination with the corresponding vector data. They can be treated like any other geometry object using the same methods. There are various ways to add styling to a document instance.

- Directly in the SVG document
- SVG content generated as the output from XSLT
- SVG content styled with CSS

The correspondence between the GML geometry type and the SVG structure is easy enough to establish as we retrieve the same geometric figure. Note that GML features do not correspond one on one with SVG features. *[27]*

Table 2-2: Difference between the syntax of Geography Markup Language and Scalable Vector Graphics.

Geography Markup Language	Scalable Vector Graphics
point	rectangle, circle or icon at the point
linestring	polyline
box	rectangle
linearring	polygon
polygon	polygon
multipoint	rectangle, circle or icon at the point
multilinestring	group of polylines
multipolygon	group of polygons

The coordinates of geometric elements in a GML dataset are relative to a Spatial Reference System (SRS), given by his name in the 'srsname' attribute. Within SVG, the coordinates can be expressed in pixel, centimeter, inch etc. A conversion to superimpose the GML space defined by the 'boundedby' property with the SVG user space is required. The generated SVG document will have a structure similar to the GML document. The 'class' property of graphical objects is needed to identify the element origin. The properties of one feature are to group with the <g> element, resulting in a distinction between various feature groups (layers or maps). The following example shows the difference between a GML fragment (2-6) and the corresponding SVG fragment (2-7).

Fragment 2-6 & 2-7: Example of a Geography Markup Language content and the equivalent in Scalable Vector Graphics syntax.

```
<Tree>
<gml:PointProperty>
<gml:Point srsName='EPSG:28992'>
<gml:coordinates>523.356,159.725</gml:coordinates>
</gml:Point>
</gml:PointProperty>
</Tree>
```

```
<rect x='523.356' y='159.725'
height='10' width='10'
class='TreePointProperty'?>
```

2.2.4 – Reasons for using GML

There are various advantages of using and creating GML documents:
- GML (and SVG) is scalable; thus not limited to a single, fixed pixel size.
- GML is an open and vendor independent standard
- GML data will be viewable and editable by a variety of software.
- Being ASCII based, any XML document can be read and edited using a simple text editor.
- XML's flexible architecture allows data being used in different ways by different browser-applications and it provides a method to verify data integrity.
- GML holds promise as an important means of storing geo-information, partly because XML's XLink and XPointer provide a means for building complex and distributed geographic data sets in which data developed for one purpose can be readily integrated with data developed for another.
- GML is based on a common model of geography, which has been developed and agreed to by the vast majority of GIS vendors in the world.
- GML used with XML non-spatial languages will enable spatial/non-spatial integration.
- SVG allows for interactivity, animation and is compatible to other XML and GML standards.
- GML is a logical way to pack spatial data for LBS involving mobile Internet devices. *[19, 40]*

2.3 – GML in LBS

Content is delivered to and displayed on wireless devices using various specialized markup languages and data formats, like:
- Wireless Markup Language (WML): part of the WAP standard
- Wireless Bitmaps (WBMP): monochrome image format, part of the WAP standard
- Handheld Device Markup Language (HDML): older standard for mobile phones
- Mini-HTML: specially formatted HTML pages hard-coded to display on small screens
- Web Clipping: Reformats a HTML page for display on small screens (no control of formatting)
- HTML: Low-end devices need clipping and low bandwidth causes restrictions

These formats emerged out of different needs and give problems when combined in one device. Except bitmaps, these formats are not capable of displaying geographical content, but are not scalable and need to be resized for different mobile devices. Therefore, the advantages of XML and GML can play an important role in displaying spatial and non-spatial data in mobile devices.

GIS software components can perform type discovery on GML schema's to identify which XML elements from the GML dataset represent a feature, a feature's properties and a feature's geometric properties. Difficulties occur when interpreting the XML elements into a geographic context and the context into a GIS systems own local context. GML helps software components interpret XML data by constraining the interpretation of the data into a well-defined geographic context. *[28]*

These GML difficulties contribute to the fact that GML is not implemented in commercial software packages yet. *[45]* More and more XML is used for display, exchange and maintenance of Internet data and there are even some websites, which make use of SVG to show geographic information. GML is hardly used in GIS packages (import and export utilities); the use of GML within Location-based Services is even more limited. Research institutes are developing their own applications and some commercial vendors, like ESRI and Intergraph are working on solutions.

A lot has to do with the limitations of mobile devices, like low bandwidth and limited storage capacity. GML is stored in readable ASCII-format and datasets are described using tags. Although it provides the ability for humans to read GML documents, the disadvantage of GML documents is that they are inherently large. To illustrate: a single laser scanning project containing 600 million points stored in lon/lat/height with 5 digits precision require 16,8 Gb disk space without tags. When converted to a proper GML document increase the amount of data by a factor of at least two. *[19]*

Various solutions are developed to handle this problem. Compression is one possibility, but it is rather time consuming and the saving is limited. Another solution could be to combine XML documents with the advantages of binary data storage (binary XML). Descriptive information is stored in a binary header, often stored as prefix of the actual dataset and the main dataset is included as binary data stream within the XML document or referenced using a link within a tag. *[28]* Another solution has been tried, is the replacement of GML constructs by a tailor-made, reduced version called compact Geographic Markup Language (cGML). Long GML tags have been replaced by shorter ones (e.g. LnStMb in stead of LineStringMember) and it reduced bandwidth and memory occupation by 60%. *[29]* Furthermore, coordinates are expressed with high precision, often more than necessary. Therefore, maps sent to mobile devices are pre-projected and pre-scaled by the server, depending on the users' device.

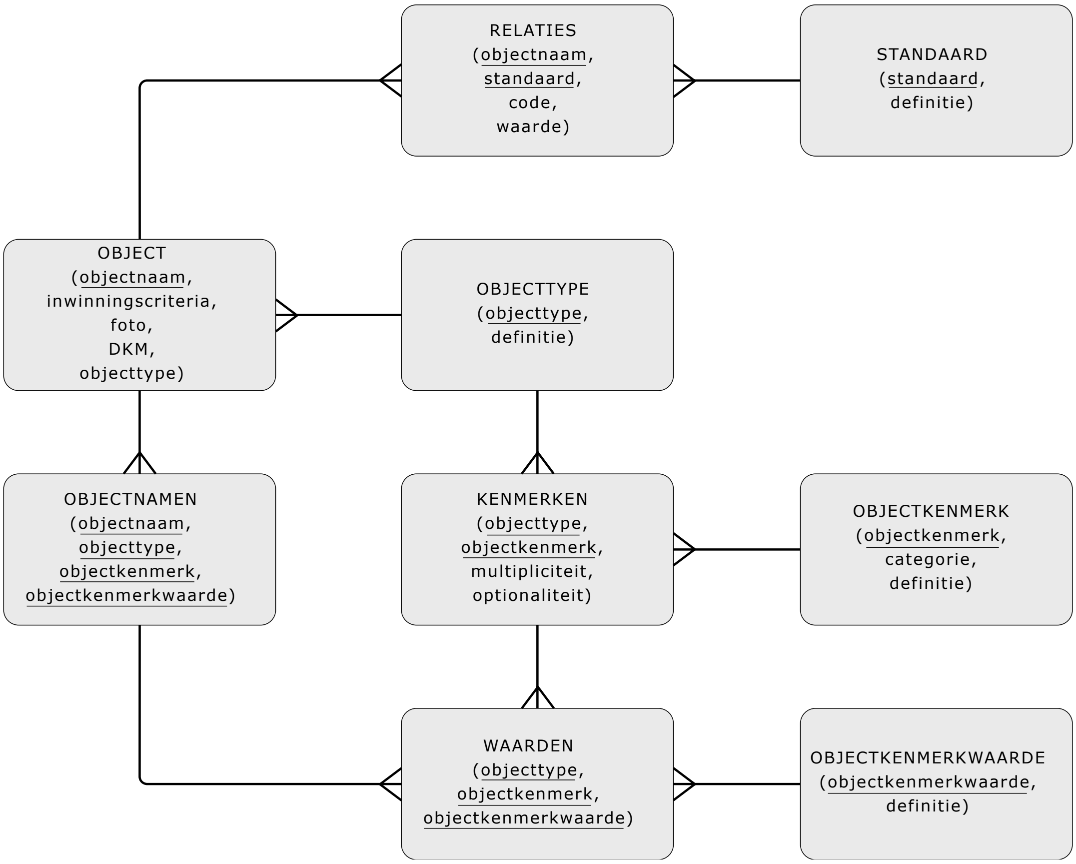
In the field of mobile GIS, there are still lots of problems to be solved. GIS companies, like MapInfo, MapGuide, Intergraph and ESRI are doing research on mobile GIS; light-weighted mobile GIS applications have been developed. These packages do not provide to full scale of functions that desktop applications have, but are made to fulfill the needs of the user in the field. But because of the limitations of wireless communication and the mobile device, it is very hard to apply mobile GIS widely. *[41]*

2.4 – Example project: Top10NL

To illustrate the growing use of GML in the geo-information market, the example of the TOP10NL project is given. Since the Open GIS Consortium recommends Geography Markup Language, not only GIS software vendors became interested in the possibilities of GML and object-oriented modeling. Also data suppliers see the potential of using open data standards in order to avoid convert processes and reach as many costumers as possible. The 'Topografische Dienst Nederland' (TDN), one of the largest geographic data suppliers of the Netherlands, has begun to restructure its data into GML. They launched the 'top10: 21st century' project, which consists of various smaller projects. These different parts consist of an inventory of the user's demand and wishes, structure definition of the Digital Landscape Model, implementation of the prototype in XML and an evaluation of the project. *[30]* The current TOP10 vector is maintained in design-files and not in any kind of database, which causes limitations. More importantly, lines and polygons do not have their own unique ID, which makes it difficult to join the files with external geo-data. *[20]* Therefore a new prototype is defined that meets the new open standard requirements and is implemented as GML by TU Delft.

Besides the conversion of the existing Microstation design-files into the GML format, the TDN wanted to make the catalogue containing all the object information publicly available via XML. All metadata of the objects is stored in a relational database. A relational database is used, because it provides an easy way of storing, maintaining and publishing metadata. Furthermore, the possibility exists to join the meta-database with the object-oriented database from the TU Delft, containing all GML objects. Before any data can be retrieved out of the meta-database, the structure is redesigned in order to obey the database construction rules.

Figure 2-4:
Entity - Relationship diagram of the database, used to make the TOP10NL object catalogue publicy available.



On top of the structure are entities and each entity has various attributes; which attributes depend on the entity. Each attribute has a certain domain (possible values), depending on the attribute. On a lower level are the TOP10 objects, like highway, canal and windmill. Each of these objects has attributes; the same ones as the corresponding entity. Also the objects' attributes have a domain. The domain values can be exactly the same as the accompanying entity or merely a subset. To make things even more complicated: certain domain values of the entity are the same as the TOP10 objects. Given these facts, the following database structure has been designed.

The database consists of four main tables, namely entity, object, object attribute and object value. These tables are connected, with or without an associative table. The table 'entity' contains unique entities and their definition. Each entity has multiple attributes, stored in the table 'object attribute' and is linked via an associative table called 'attributes'. It contains each unique combination of entity and object attribute. Furthermore it contains multiplicity and whether it is optional or not. Every unique combination of entity and attribute has a certain object values, stored in 'object value' and linked via the associative table 'values'.

The table 'object' contains every unique object and each object belongs to one entity. More object information is stored in this table, like picture and Digital Cartographic Model (the way the object is displayed). Furthermore, each object has attributes and object values, which are connected to 'values' in a new associative table 'object names'. Finally, two tables are added to link to definitions of the TOP10 objects with the definitions of other object standards, like the NEN3610 or DIGEST.

In order to retrieve data out of the database, it is not necessary to include the table 'values'. By using a query, the same data can be collected. However, by using a query, the internal control of the data will be lost. Only disadvantage is additional work, when filling the database. Maintenance of the database does not require much expertise and knowledge of the structure, especially when the options 'enforce referential integrity' and 'cascade update referenced fields' are enabled. These options ensure that changes in the four main tables are carried through in the entire database.

Because the TDN wanted to make the object catalogue publicly available via XML, the project is not yet fulfilled. Additional investigation is required, but the basis is accomplished.

2.5 - Summary

Location-based Services are emerging by the day and especially in commercial sectors, large profits are being made. In the GI-sector mobile services are developed, but since GIS applications demands more from hardware and software, development of professional mobile GIS services are slowed down. Recent GML developments have proven to be suitable for the storage and transfer of geographic objects and offer the advantage to move beyond static raster images. The advantages of scalability, vendor independency and human readability seduce spatial data users and suppliers to use GML more and more. Scalable Vector Graphics are often used to display GML data, since it is part of the same Markup Language group, but SVG viewers for mobile devices are limited in number and in capabilities. An additional disadvantage of SVG for professional applications, are the lack of functionalities needed to be a fully operable service.

The attempt of this research project is to incorporate the GML data format into an application for inventory and object update purposes, which is suitable of supplying Location-based Services. LBS are evolving rapidly, new technologies emerge, but these developments focus mainly on hard- and software and less on (spatial) data content. Seeing the advantages and expectations of GML and Location-based Services, GML and LBS seems to offer a promising combination.

3 – LBS design

The world around us is constantly changing. Buildings, roads and objects are added, removed or changed and all these changes have to be recorded: into the field with a map and a piece of paper, register changes and add these into the computer back at the office. Would it not be easier if a user only has to carry a mobile device and enter the changes just once?

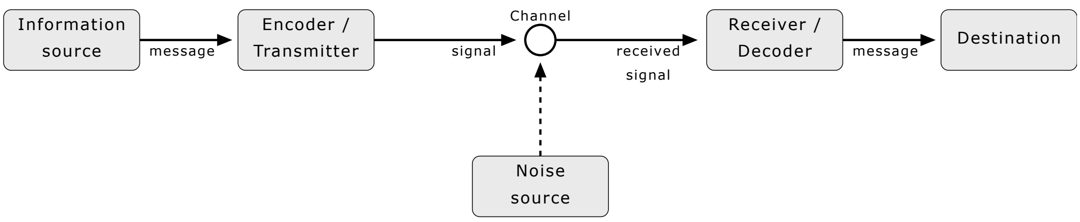
LBS are characterized by their ability to support itinerant (while moving with a user), distributed (integrating functions that are performed at different place, transparent for a user) and ubiquitous (delivering the same functionality independent of the user's location) computing (see paragraph 1.1). A suitable Location-based Service to investigate the use of GML on mobile devices is a location-update application. It can be used in the field, to add, update or remove objects, but it can also be used for inventory purposes.

3.1 - Communication model

A way to describe the desired LBS is the Shannon – Weaver communication model. Claude Shannon and Warren Weaver have produced a general model of communication in 1947. The model consists of six elements:

- Source
- Encoder
- Message
- Channel
- Decoder
- Receiver

Figure 3-1: Communication model by Claude Shannon and Warren Weaver.



According to their model, a message begins at an information source, which is relayed through a transmitter and sent via a signal towards the receiver. But before it reaches the receiver, the message must go through noise (sources of interference). Finally, the receiver must convey the message to its destination. *[42]*

3.1.1 - User Requirements

However, before any design can be made, desired functionalities have to be determined. A Location-based Service for the maintenance of topographic objects must be able to:
- Provide the user geographic data, based on its current location
- View, change and add location information of objects in the real world
- Transfer altered information back to central storage location
- Embed GML in the exchange process

3.1.2 – Model translation

When the desired service is translated to the communication model, the elements will look as follows:

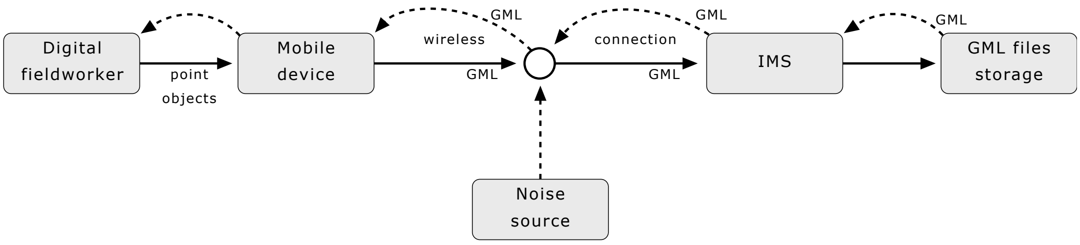


Figure 3-2: The translation of the Location-based Service design into the Shannon-Weaver model of communication. The user sends a GML containing information about objects to a server.

Source of the model is the digital fieldworker, who gathers location data of point objects. The data is entered into an application on a mobile device (encoder); new objects can be added, but also update or removal existing ones should be possible. By using a GPS receiver, the coordinates of each location are captured. Maps of the surroundings can support the user by showing its position, relatively to other objects. An Internet Mapping Server (IMS) provides this additional spatial data.

The gathered information is stored in a GML document (message), which is being send from transmitter (mobile device) to receiver via a wireless Internet connection and at the same time, the GML document displays the features on the mobile device. The GML document, containing the data gathered by the fieldworker, is received and stored on a central server, where it becomes available for other employees, but also third party users can use the data (destination). The use of the collected data by external users is possible without any conversion, since GML acts as exchange format. Especially when multiple fieldworkers are involved, it becomes important that the collected data becomes immediately available. Every fieldworker can have access to the most up-to-date information by importing the GML document from the server to its mobile device (the data flow in the other direction).

3.2 - OGC specifications

To improve the interoperability of the service; it is important that the OGC specifications are being used as much as possible. GML is one of the specifications, which is being incorporated into the LBS, but the Web Feature Service (WFS) and Web Map Service (WMS), which are part of a Internet Mapping Server, are other important specifications.

A WMS is capable of producing maps drawn into a standard image format (PNG, JPEG) based on a standard set of input parameters. When pixels in the resulting map contains no data, they become ‘transparent' and thus several independently drawn maps can be laid on top of each other to produce an overall map, even when maps come from different Web Map Servers. A disadvantage of using raster data is the lack of comfort of handling and various cartographic aspects (e.g. fonts). Also, a server contact is necessary for each client’s request.

The interoperability that enables this automatic map overlay comes from a set of common interfaces for communicating and a few basic commands/parameters. This set of interfaces is known as the Open GIS Implementation Specification and includes the Web Map Server interface Implementation Specification. These specifications address basic Web computing, image access, display, manipulation and coordinate transformation capabilities. To put it in other words, they specify the request and response protocols for open Web-based client/map server interactions.

In contrast to WMS, a Web Feature Service (WFS) produces feature services. It allows users to retrieve spatial data encoded in GML from multiple Web Feature Servers. With the appropriate infrastructure a web feature service can deliver spatial data directly into a user’s desktop GIS application, removing the requirement to manage any local repository of data. Lower data volume transmits faster and can be handled by a standard web browser (with plug-in). However, vector data is manufacturer dependent and data volume can differ per area. Chaining of data and Geo-spatial applications could be vital in the delivery of successful Location Based Services, where the end users terminal has limited processing and display capabilities. *[37]*

OGC compliant WFS allow clients to retrieve geo-spatial data encoded in Geography Markup Language (GML) from multiple Web Feature Services. The requirements for a WFS are:
- The interfaces must be defined in XML.
- GML must be used to express features within the interface
- The filter language will be defined in XML
- The storage of geographic features should be transparent to client applications and the only view of the data should be through the WFS interface
- The use of a subset of XPath expressions for referencing properties

To support transaction and query processing, the following operations are defined: GetCapabilities (Indicates which feature types it can service and what operations are supported on each feature type), DescribeFeatureType (Describe the structure of any feature type it can service), GetFeature (WFS must be able to service a request to retrieve feature instances, constrained spatially and non-spatially), Transaction (WFS may enable to service transaction requests, composed of operations that modify features) and finally LockFeature (WFS may be able to process a lock request on one or more instances of a feature type for the duration of a transaction, to ensure serializable transactions). *[43]*

3.3 - Summary

The Location-based Service design as described in paragraph 3.1.2, has to be translated into components of an application, built-up out of hard- and software components, which is capable of supplying the designed LBS. The implementation of the application aims at using the OGC standards and open source solutions if possible.

4 – From LBS to application

Figure 4-1:
A small collection of mobile devices, which are capable of receiving Location-based Services, but not all are suitable for use of a professional object maintenance application.

In order to have a successful application, the various parts of the architecture have to be well chosen. Location-based Services do not differ from desktop applications; they have to meet the requirements of its users. LBS technology can be divided into four categories, namely client terminals, network infrastructure, application servers and data. Recent years, rapid development has been evident in the areas of client terminals, network infrastructure and application servers, but developments in the area of data access have been somewhat overlooked in association with LBS. *[31]*

In the previous chapter the requirements and design of the Location-based Service were determined. In this chapter the design is being translated into an application, capable of supplying the LBS. The various components of the application are explained and comment will be given on the choices that are made.

4.1 – Client terminals

Many mobile devices, able to make use of LBS are available, but most of them are not suitable for running an application. All phone-based devices are mainly used for 'where is the closest' service requests. With the arrival of next generation mobile devices, the possibilities extended; via an operating system, software can be installed and hardware components can be attached to the device. They are known as 'smartphones' and are smaller than a notebook or pen computer, becoming more attractive for outdoor employees. However, storage capacity and processing speed is limited and software has to be kept minimal in order to function properly (thin client).

4.1.1 – Software

Special light-weighted GIS applications have been designed to run on mobile devices and are suitable to use, since they contain most desired functionalities. These functions consist of positioning, data request and data manipulation. However, none of the available GIS applications for mobile devices support the vector-based Geography Markup Language as format on the mobile device itself. *[45]* Intergraph LocationServer and ESRI ArcPad have been under investigation, but both do not fully support GML functionalities. LocationServer provides the ability to read and write to a GML file, which is stored on a central location, but uses a different file extension (.ond) as exchange format. This format is optimized for use on mobile devices (e.g. high compression of data). Also beta mobile SVG viewers are available. SVG viewers are capable of displaying point, lines, polygons and text string and it is capable of zooming, panning and displaying properties. Furthermore, SVG provides interactivity, like adding data layers, showing attribute data of an object and even features can be added manually. But a SVG viewer is not a GIS application, resulting in shortcomings when developing an application. Some of the shortcomings are:

- Support of GPS receivers
- Adding of external data layers
- Sending acquired data towards a server

These software drawbacks are strengthened by hardware limitations of a mobile device. Thus SVG can be used for display purposes of gathered data, but not as programming environment. Although LocationServer and ArcPad do not support GML on the client side, it this does not mean GML will not be implemented in the application. Via a database, it still is possible to make GML data available for other users and third parties on the server side of the application.

ArcPad has been chosen to function on the mobile device (the investigation of LocationServer was hampered by difficulties during installation). ArcPad already contains most of the required functionalities, like connecting with a GPS receiver, displaying geographical content and adding point objects. Another advantage is the availability of ArcPad Studio. It is additional software to be installed on a desktop computer, which offers the opportunity to customize and create ones own ArcPad application. ArcPad uses XML to store toolbars, forms and actions that are part of the customized application. In theory, the application can be created without the help of ArcPad Studio if the XML syntax is known, although it will not be easy.

4.1.2 – Positioning techniques

In order to request location specific information, the position has to be known. Several positioning techniques are at hand, but not all are suitable or cannot be used.

Table 4-1: Various positioning techniques exist, each has a different spatial accuracy, which is a criteria whether a technique is usable.

Technique	Spatial accuracy (m)
Cell of Origin	100 - 5000
Time of Arrival	50 - 150
Enhanced Observe Time Difference	5 - 50
Global Positioning System	1 - 10

Cell of Origin method is the most straightforward solution and uses cell identification information within the mobile telephony network to identify the approximate location of the caller. The accuracy differs largely, 100-500 meters in urban areas, but as low as a few kilometers in rural areas. Time of Arrival (TOA) is a commonly used network-based solution for determining the callers' position. The differences in the time of arrival of the signal from a user's mobile device to at least three base stations are used to calculate the location. The Enhanced Observed Time Difference (E-OTD) technique determines the location of mobile devices by using location receivers, which are geographically dispersed across a wide area. It requires new handset capabilities and software. The accuracy ranges between a few meters up to 50 meters and depends largely on the operational environment (sky obscuration, indoor). The main drawback of terminal based solutions is that they require new handsets.

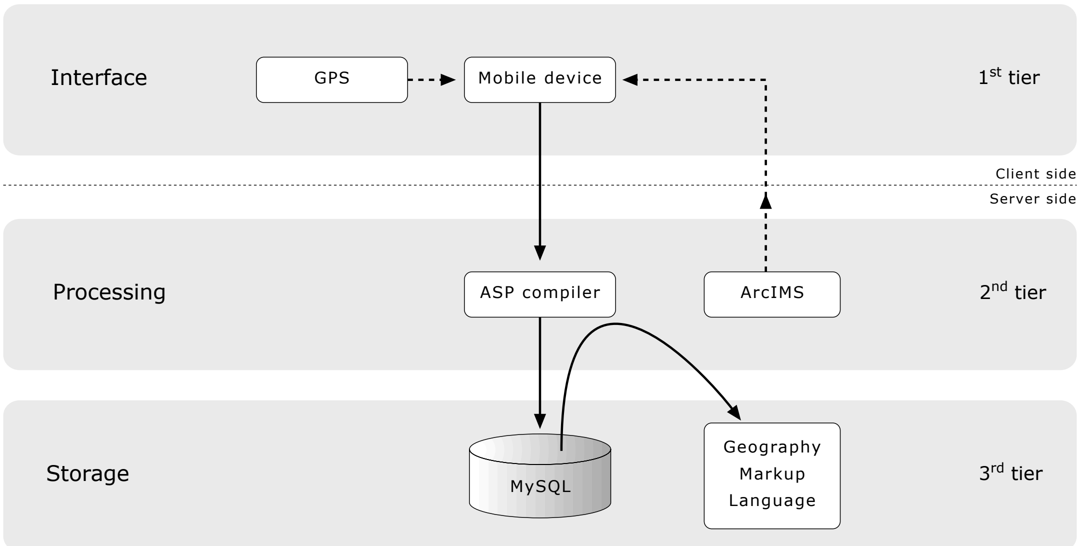
GPS is a navigation system that uses a constellation of satellites, circling above the earth. Like TOA and E-OTD, GPS is a time based positioning method. The receiver searches for satellites and decodes the information transmitted, enabling to perform the positioning. With three satellites XY coordinates can be calculated and with one satellite more, a 3D position can be estimated. It GPS is accurate, within meters, but has no indoor coverage. It can take some time to fix unless it is permanently tracking satellites, which in turn may require significant power consumption, which can be a serious limitation for handsets. *[10, 33, 44]*

The Global Positioning System is chosen for the application, since it a well-known and widely used positioning technique and suitable, because of its availability and relatively good accuracy. Various GPS receiver are on the market and can easily be attached to the mobile device.

4.2 – Application server

Secondly, a server is needed where most of the processing should occur in order to limit to processing on the client side (thick server). The server side application consists of three parts: (spatial) data storage facility, compiler software, enabling access between the smartphone and the database and thirdly, a web map server to provide the user with additional geographic information via maps.

Figure 4-2:
The LBS design is translated into an three tier application architecture:
- Graphical User Interface (GUI)
- Processing
- Data storage



4.2.1 – Compiler software

The architecture can be divided into three tiers (see figure 4-2). The first tier is the user, equipped with a mobile device. Second tier is the software, which processes the outgoing and incoming data on the server. Active Server Page (ASP) is a scripting language used for communication between input and output on the server side. It is a well-known language (together with PHP) on the Internet to keep track of the user's privileges, user's input and for interaction with a database. ASP is developed by Microsoft and has to use Microsoft Internet Information Service (IIS) as server software in order to function, which could be a disadvantage. In contrast to ASP, PHP is an Open Source Language, but both languages can be used for communication with databases. Since the server is running Microsoft Windows 2000 Server, using ASP will not be a problem.

In order to get the location information, gathered by the digital fieldworker in the database, ArcPad sends the data to the server via an URL string. It consists of an URL location and multiple variables, including the GPS coordinates. The ASP page reads the variables out of the URL string and sends them in a SQL statement to the database, where they are stored in a table. ASP can be used with any database, as long as there is an ODBC connector available.

4.2.2 - Database

The third tier of the application architecture is the storage of (spatial) data. Often GIS application store spatial data and non-spatial (attribute) data separately. This split up has several drawbacks, as it is difficult to maintain data integrity between the spatial and the non-spatial part, because the same database engine does not manage the data. Benefits of storing both parts in the same database are better data management, reduced complexity and a seamless integration of GIS data stores. *[34]* Conventional relational databases can store spatial data as plain variables, but cannot execute spatial queries, because geometry, which is the key aspect of spatial data, is not supported. Any database system attempting to deal with spatial applications has to provide the following features:
- A set of spatial data types to represent the primitive spatial data types (point, line, polygon and operations on these data types (e.g. intersection, distance)
- Spatial types and operations should be part of the standard query language that is used to access and manipulate non-spatial in the system
- The system should provide performance enhancements, like indexing and parallel processing

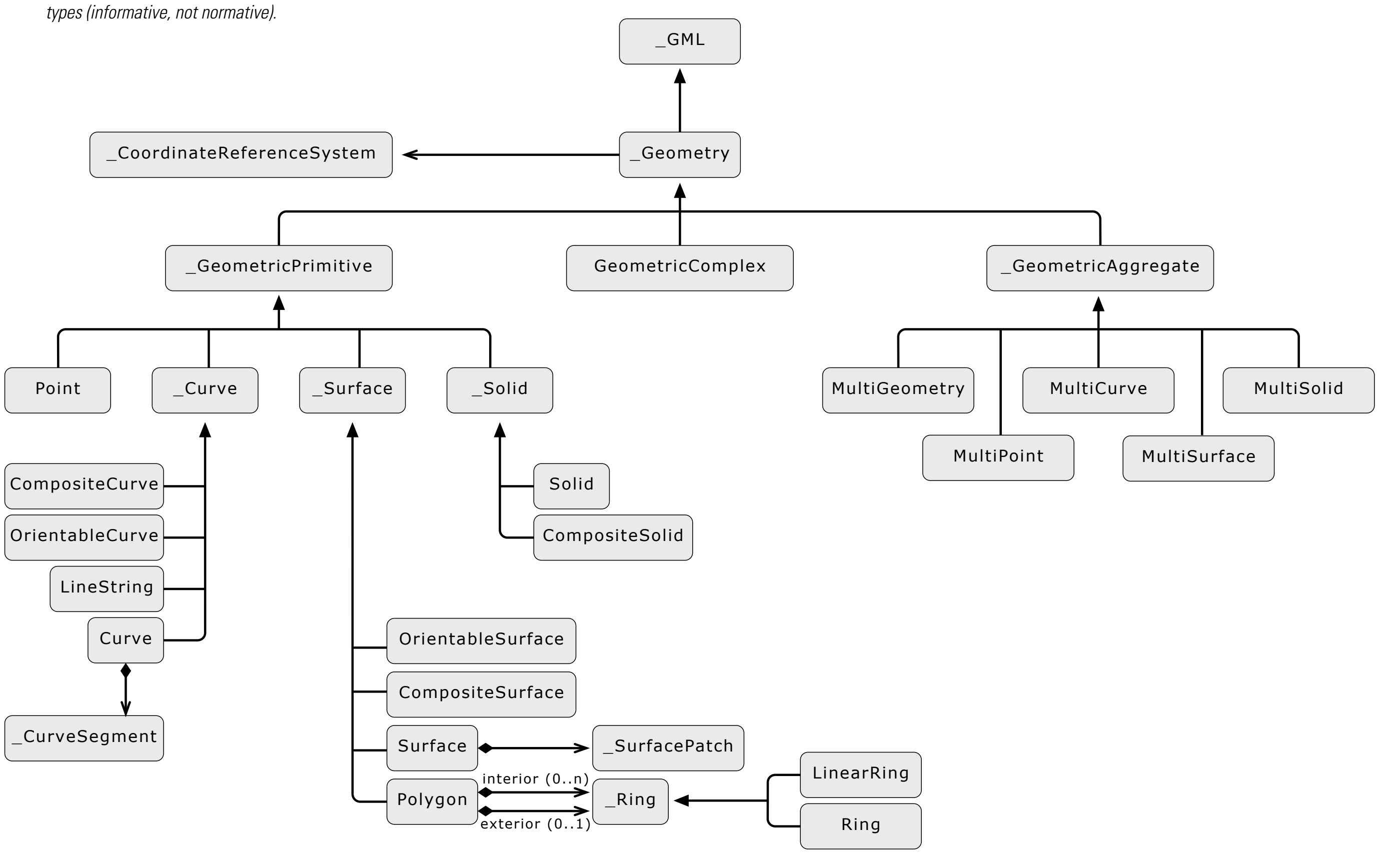
Since the research project focuses on the use of GML in Location-based Services, only point objects are being gathered in the field. Collecting lines and polygon objects would be possible, but requires adjustments on both client and server side. As a result, this project makes use of a relational database, because no spatial queries will be performed on the database and the geometry is limited to three coordinates (XYZ), which can be stored as attributes in a normal table. Storing lines and polygons as an attribute on a relational database is more difficult. The number of coordinates for each line or polygon is not fixed and can amount to hundreds of points; multi geometries (see figure 4-3) are even more complex. A trick to use relational databases for more complex geometry types is to take the geometry out of the data file and turn into a long string of coordinates, including the geometry type. This notation is also used to store geometries in GML documents.

*Firagment 4-1:
A way to store line or polygon information into a relational database: as one long string. A GML document also stores line and polygon coordinates in such a away.*

```
POLYGON((173103.06250718 443070.68749710,173129.73441120 443016.28128905,173152.06250722
442970.12500101,173170.60938724 442932.50000897,173175.12500325 442922.74999296,173178.593
70725 442915.28128895,173201.54689127 442867.46871291,173208.39060328 442852.37498490,1732
11.18749928 442846.18749689,173216.81249128 442833.78128888,173234.37498730 442798.0000088
5,173240.20311531 442784.78128883,173243.35940331 442777.62500083,173245.97309931 442771.70
058482,173232.49249130 442771.70058482,173211.43751528 442815.50000886,173179.26561125 4428
84.21879293,173172.85940324 442899.59370494,173170.60938724 442910.65629695,173103.06250718
443070.68749710))
```

In this project, two types of relational databases are used, namely Microsoft Access (part of the Microsoft Office suite) and MySQL. MySQL has several advantages, since it is better capable of dealing with large datasets and secondly it better copes with multiple users. Furthermore, it is Open Source, which contributes to goal of using open source software. MySQL is often linked with PHP (probably because both are Open Source), but also ASP can be used. It can be downloaded *www.mysql.com* and is licensed under the GNU General Public License (GPL). Another advantage (not relevant for this research project) is that version 4.1.0 has a built-in spatial extension; therefore suitable for generation, storage and analysis of geographical data. More importantly, the implementation of the spatial extension complies the Simple Features Specifications by the Open GIS Consortium. MySQL implements a subset of the SQL with geometry type's environment proposed by the OGC. A geometry-valued SQL column is implemented as a column that has a geometry type. The specification describes a set of SQL geometry types, as well as functions on those types to create and analyze geometry values. However, MySQL with spatial extension is still in its alpha stage, thus not a fully reliable production release. The full set of OGC Specifications is not yet supported; especially the Spatial Reference Identifiers (SRID) support is missing.

Figure 4-3: An overview of the type hierarchy of the Geography Markup Language geometry types (informative, not normative).



The only issue advocating for the use of Microsoft Access as relational database is its portability. An Access database consists of one single file, which can copied and transported easily. MySQL has to be installed and run as a service on the server. It can completely be controlled using the command line, but also a Graphical User Interface (GUI) is developed, enabling the user to control the database server with a more 'windows-look'.

A relational database is still a table containing variables and exchange of spatial data towards third parties and other users is not yet established, but Geography Markup Language provides the solution. By using ASP scripting, data can be taken out of the database and stored in GML documents. The produced GML files can be used in various ways, as exchange format or converted to SVG for display purposes. GML means flexibility, exchangeability and scalability; Third parties can benefit from GML, without any conversion or predefined rendering and can be used for display purposes in various media.

4.2.3 – Web Mapping Server

Besides ASP scripting, an Internet Map Server (IMS) is part of the middle tier software. It is capable of distributing GIS information over the Internet. An IMS is needed to provide the user with additional geo-information of the surroundings while standing in the field. It converts spatial information, available as services in the server, to the requested format and sends it towards the mobile device. A software solution, which is available, is ArcIMS by ESRI. It gives the advantage that it can easily be used in combination with ArcPad, since both a developed by the same manufacturer and can provide both map and feature services. If an IMS layer is being added, a XML request is send to the server and the response is send back, also wrapped in XML (ESRI calls it ArcXML).

Limitations concerning the display of vector-oriented graphics, mentioned in paragraph 4.1.1, also account for WFS, provided by the IMS. ArcPad is not capable of handling features services; a drawback of the capabilities of the software. A remark has to be made concerning the terms WMS and WFS. ESRI uses these terms for specifying their map and feature services, but this does not mean these services are OGC compliant. Only recently, ESRI has launched WMS and WFS connectors for ArcIMS, which incorporate the OGC specifications. These connectors are not being implemented in the application architecture, since the release was halfway this research project.

4.3 – Network infrastructure

The server and client have to be connected in order to benefit of the Location-based Service. Such a network is quiet simple; only a wireless connection is needed between the smartphone and the server. Various techniques are available and although data transfer will be limited, a larger bandwidth is desired. More types of communication protocols for wireless Internet are available: Global System for Mobile communications (GSM), General Packet Radio Service (GPRS) and Universal Mobile Telecommunications System (UMTS). *[33]* Nowadays, most mobile devices are equipped with GPRS, which still have a limited bandwidth. Besides the bandwidth, the stability is important; because the collected point objects are directly send towards the server.

4.4 – Data

The primary use of the designed Location-based Service is for maintenance and inventory of point objects in the field, is based on large scaled data. For the application, sample data of Wageningen would be sufficient, saving time on data gathering and preparation. Topographic data will be displayed via the web map server. Various data of Wageningen is available; suitable for its purpose are shapefiles on scale 1:10.000, originating from the TDN, which are imported as a service on ArcIMS. The LBS is suitable for users in different fields of expertise, like Forestry, Water management and soil science, because merely every dataset can be used. However, when users conduct an inventory in the field, only a limited area should be requested from the server, mainly because of the shortcomings of mobile devices.

Figure 4-4:
Since it is a wireless application, any area will do as testsite, as long as their iare maps available of the surroundings and a GPRS connection can be established. For testing purposes, a small area of Wageningen is sufficient.

4.5 - Summary

One important conclusion can be drawn out of this chapter. The lack of Geography Markup Language support by the mobile device and ArcPad is an important restriction to the original LBS design. Nevertheless, via the use of a database, the application is able to provide the user with GML files on the server side.

The application design contains to following components:
- A mobile device (equipped with a GPS receiver and a wireless GPRS Internet connection) containing the ArcPad application. The user will be able to request additional data for orientation and background information, which is send via the ArcIMS map server.
- Point objects can added by the user and altered information will be send to the server via an URL string, where the variables are read by ASP. This page will send a SQL statement to the MySQL database, where all the spatial and non-spatial data are stored.
- The same ASP compiler transforms the contents of the database into a GML file, according to the accompanying XML Schema Document and becomes available for other users without any conversion or transformation.

5 – Application implementation

The Location-based Service design of chapter 3 has been translated into application components and now the application design has to be implemented. In this chapter, the application design will be turned into a working application, describing all the components and settings and how they interact with each other. The LBS is tested in the field and the findings are described in paragraph 5.3.

5.1 – Application description

The basic idea of the original LBS design, of chapter 3 still stands. All user requirements (paragraph 3.1.1) are present, but the way the LBS design is been implemented differs from the original idea. Mainly the GML component has been embedded differently, because of the limitations of the smartphone.

The user can request Web Map Services from ArcIMS while standing in the field. Since the GPS receiver is active, the correct part of the map is send to the mobile device via GPRS. Then, the fieldworker can start editing the various locations. Via the ArcPad application the user can add new locations by adding attribute information of a location to a form. After the form is closed, the attributes and the GPS coordinates are send to the server in an URL string. The position will also be added to the map, which lies on top of the WMS layer. Besides adding a new point, the user can update or remove an existing location. This is done by entering the ID of the existing point in the form and indicating whether this location should be updated or removed.

Multiple users can take part in the inventory process, at the same time or at different times. The fieldworker does not have to remember the ID of a location, because the maximum ID of a location is taken out of the database. So, the next point, which will be entered has the maximum ID + 1. It could occur that different users want to add a location with the same ID. Then, the principle of first comes, first serves takes in, since the ASP compiler permits the update or removal of a location, which has not been added by the same user.

On the server side, the ASP compiler reads the URL string with the variables, determines whether the location should be added, updated or removed and sends to correct SQL statement to the

MySQL database. The SQL statement is executed and the table will be altered. The other important function of the ASP compiler is to takes all the data out of the database and write it to a Geography markup Language document. The document conforms the XML Schema, which is been made and accompanying the GML document. The document is available for other users and can be used without any conversion (as long as the software supports GML / XML content).

Figure 5-1: The o2 XDA with a separate Garmin e-trex summit GPS receiver are used to test the mobile application.

5.2 – Hardware

5.1.1 – Mobile device

For this research project, an O2 XDA smartphone is used, because of its size, it has a built-in GPRS connection and has good overall performances. A GPS receiver is built-in, thus a separate receiver (Garmin e-trex summit) is used, which can be connected directly to the device via a cable. It uses two separate AA-batteries, which is an advantage compared to built-in GPS receiver, because else the power supply of the XDA will be used (and the fieldwork has to be interrupted to recharge batteries).

Component	Specification
Device	O2 XDA
CPU speed	200 MHz
Storage capacity	64 Mb
Display	240 x 320 pixels
Operating system	Windows CE
Wireless connection	GPRS (provided by O2)
GPS receiver	Garmin e-trex summit

5.1.2 – Server

At the server side, most of the processing will occur; it is a necessity to reduce processing speed on the XDA. Hardware is easy to find; merely every desktop computer can be used and act as a server. Main requirements are an Internet connection and server software must run (be aware of firewall settings). Internet Information Service is running on the server terminal, which can easily be installed on computers using Windows as operating software.

Table 5-2: Specifications of the server hardware used in this research project

Component	Specifications
Device	
CPU speed	
Storage capacity	
RAM	
Operating system	Windows 2000 server
Server software	Internet Information Service 5.0 (IIS)

5.3 – The application at user level

Now that all components are explained and the architecture is described, the way the LBS functions is explained. The figure 5-2 describes the actions of user and/or application in four steps.

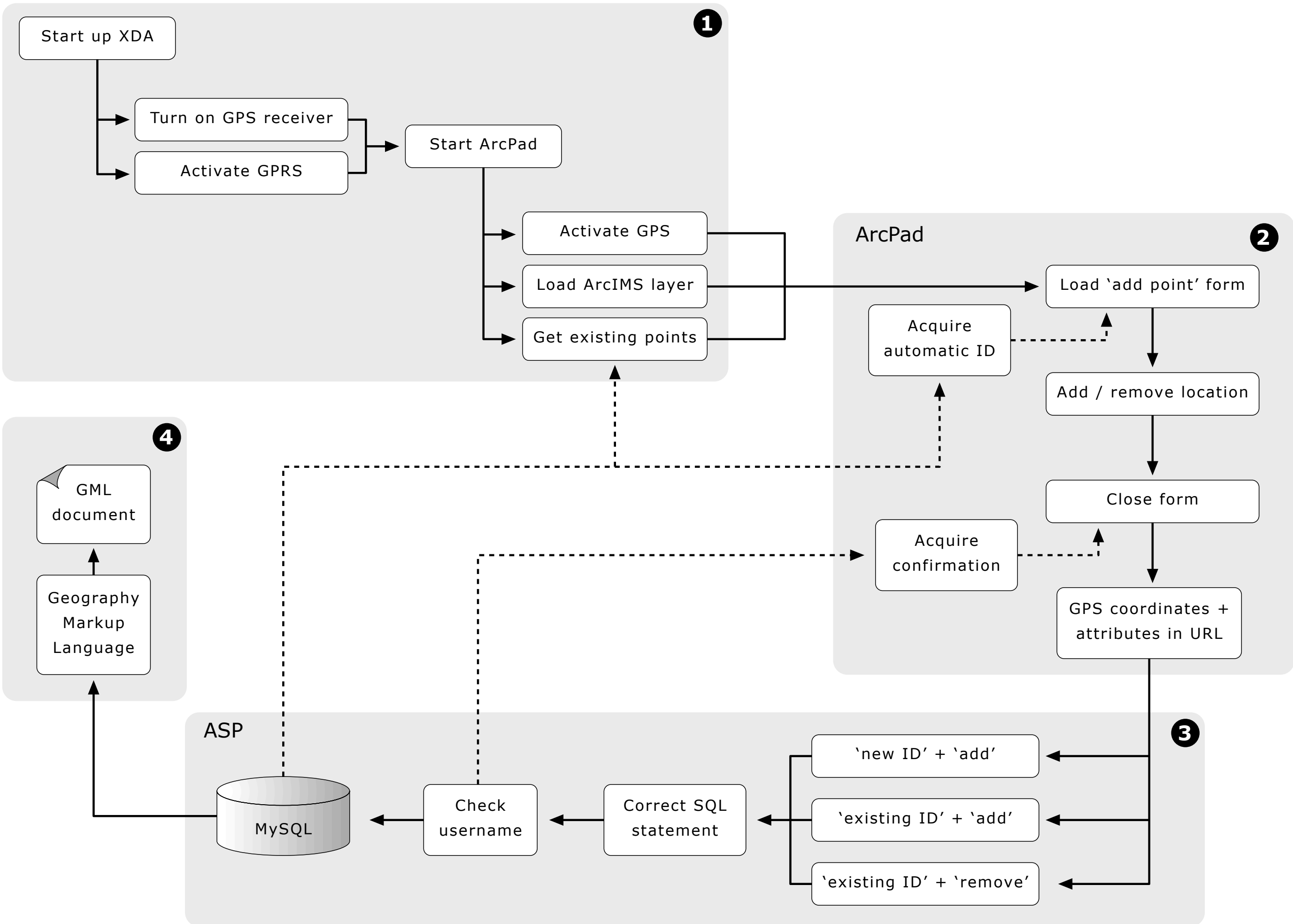


Figure 5-2: A flowchart describing the various steps the object-maintenance application and the user are taking.

5.3.1 – Starting up the application

- After the XDA is turn on, GPRS connection is functioning and the GPS receiver is sending NMEA to the smartphone, ArcPad can be started.
- The objects, which are gathered in an earlier stage, have to be transferred from the database to the mobile device in order to know which locations are already investigated. The objects can be imported by clicking on 'get points'. A request is send to the web server and via a SQL statement all coordinates and attributes are taken out of MySQL and send to the XDA, where they are stored in the new created shapefile. In case no data is present only a new shapefile will be created. Especially when multiple users are involved, this step is important.
- Additional spatial information can be imported from ArcIMS, which is needed for determining the position in the field. WMS automatically sends the correct information, since the user's position is known by GPS.

Figure 5-3:
Screendumps, clarifying the various the application and the user are taking.

A welcome form is displayed first.

Additional WMS layers can be added.

Identify shows a feature's attribute data.

The GPS shows the user's position.

5.3.2 – Adding locations

- When a user is at the correct location, the 'add GPS point' button has to be clicked. The appearing form contains a number of fields, where non-spatial information of the location can be filled in. For this research project, a limited amount of attributes are available, but can be extended easily. The form contains:
  - Theme combo box — Specifies the name of each feature member in the GML document
  - ID edit box — Provides the unique identification number of each object / location
  - Name edit box — Provides the name of each object or location
  - Add checkbox — When checked a object or location will be added or updated
  - Remove checkbox — When checked a object or location will be removed

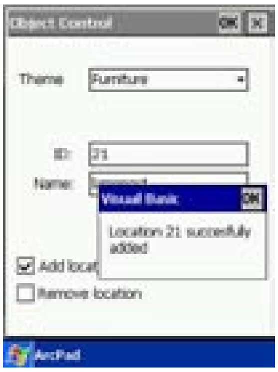
- The ID is already filled in, once the form opens. This number is taken out of the database and represents the maximum ID in the database + 1. In this way, it becomes easy to continue with an existing inventory, even if multiple users are conducting the inventory. Also, users are not bothered remembering the previous ID (unless the ID order is not sequential).
- To add a location, the 'Add' box has to be checked. However, when an existing object has to be updated or removed, the corresponding ID has to be filled in, in combination with 'add' (to update) or 'remove'. An important remark has to be made concerning the updating by multiple users. If more users are trying to add a point with the same ID, the principle of first comes, first serves takes in. Only the person, who adds the point to the database, is allowed to update this location. The ASP compiler compares both username and username of the existing object and when they do not match, the action will be cancelled. When the form is closed via the OK button, the attributes and the GPS coordinates are send to the server in an URL string via the GPRS connection and the location is displayed on the map.

5.3.3 – Storage in the database

- The ASP compiler analyzes the URL string, containing the coordinates and attributes and every attribute and coordinate is stored in memory as a variable. Then, three scenario's can occur:
  - Add location (checkbox 'add' + new ID)
  - Update location (checkbox 'add' + existing ID)
  - Remove location (checkbox 'remove' + existing ID)
- The ASP form determines which scenario's applies to the user's choice based the values of the variables and each scenario corresponds with a SQL statement. The appropriate statement is send to the database and executed.
- Since only the user who entered the object is allowed to update or remove that location, username and current date are added as additional information. Username is determined by the name of the mobile device (hostname) and the date is the system date of the server, which is acquired in the ASP form. The result is an altered database, which colleagues can access.

5.3.4 – Constructing the GML file

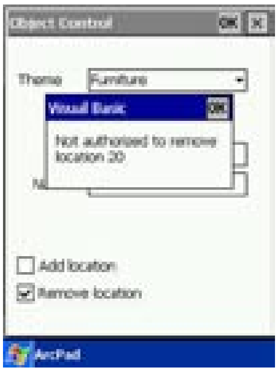
- After the spatial and non-spatial data has been stored in the database, a GML file is constructed. The ASP compiler writes the various parts of GML to an ASCII text file with .gml extension, obeying the Geography Markup Language content rules.
- Firstly, general information about the GML document and namespaces (paragraph 2.2) is written



The location is added to the database.

to the GML-file. Secondly, the spatial extent (bounding box) of the dataset is determined by looking at minimum and maximum XY values. Then repetitive elements are read from MySQL and stored in the document, placing the variables at the correct position in the document. Every theme becomes a separate feature member, belonging to the same feature collection. Finally, end tags close the start tags from the top section and the document is saved on the server, where it is available for further use by third party users.

- Besides the GML content document, a GML Schema must be available in order to make further use of the data. The corresponding Schema is made in advance, since the structure of the GML documents does not change when the content is altered and is available in the same directory.

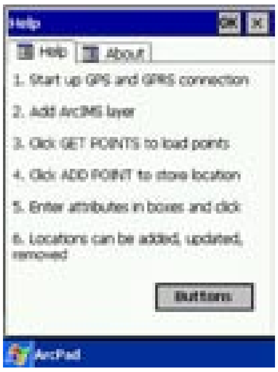


Database is locked for 'unknown' users.

5.4 – Testing the application

After the application components are implemented, it is tested in the field; the functionalities and behavior of the service are evaluated. Activating the GPRS connection and GPS receiver not always go smoothly, because of errors in the hardware settings. Sometimes, the default settings have to be adjusted in order to function properly.

The application itself functions properly, except the automatic retrieval of the ID. When the 'add point' form is opened for the first time, the ID is filled in correctly (maximum ID + 1). However, when a second location is being added, the same ID appears. It seems that the second ID does not replaces to first one, once it is set into memory. Even more peculiar is the fact that this function does work on the computer of the technical supervisor. So, the error is not in the application itself, but in other software settings (e.g. operating system).



Help can provide extra information.

The Garmin GPS receiver is functioning well and spatial accuracy in horizontal plane ranges between 5 and 12 meter, but in built-up areas the accuracy is decreasing, caused by distortion and reflection of large objects and buildings. Height measurements are less accurate and unpredictable.

The transfer speed of the GPRS is somewhat disappointing. The average transfer speed is 40 Kbit per second. For sending the location information to the server, it is acceptable, but loading ArcIMS map services takes much time and often reloads (e.g. after an object is added to the map). Of course, the user can move to the next location, while the map is loaded. The lack of Feature Service was as anticipated, because WFS is not supported on commercial mobile GIS software (paragraph 3.2.1). Sending of the URL string, containing the coordinates and attributes is working fine and the variables are stored in the database properly. Also generating the GML file does not give any problems and with the accompanying GML Schema document, the GML file can be used for further processing.

5.5 - Summary

Although GML cannot be incorporated in the client side of the application, the Location-based Service application does make use of GML successfully. All of the components of the architecture interact properly and the data is correctly stored in the database and transferred to a GML document. The application is working conveniently, but the loading of ArcIMS layer does slow down the application, due to the speed of the GRPS Internet connection. Only the automatic loading of the ID out of the database seems to function not properly. The used architecture is a successful implementation of an inventory and object-update application.

6 – Conclusions

Before any conclusion can be drawn, findings are compared with advantages Geography Markup Language claims to have, as written in paragraph 2.2.4, especially last three items are important for this project:

- *GML (and SVG) is scalable, not limited to a single, fixed pixel size.*
  GML and SVG make use of vectors instead of raster, which is an advantage, since it allows unlimited zooming without loss of details (see figure 2-3). Especially on mobile devices, zooming is often used, due to a limited screen size. However, a comparable effect can be simulated using only raster images, like the use image pyramids, although the raster images are often larger in size.

- *GML is an open and vendor independent standard*
  The use of GML ensures a worldwide support, since the Open GIS Consortium recommends it. XML and GML can be used freely, because it is an open standard.

- *GML data will be viewable and editable by a variety of software.*
  Before any GML software is developed, the developers want to know the potential of GML. The lack of GML support, especially on mobile devices, shows that software vendors are waiting until GML is widely introduced and available. Ultimately, the users determine the success of GML, not the manufacturers.

- *Being ASCII based, any XML document can be read and edited using a simple text editor.*
  XML and GML have to advantage that it is both machine and human readable, making it suitable for a simple text editor. But knowledge of the language is necessary; especially GML has a complex and extensive syntax.

- *XML's flexible architecture allows data being used in different ways by different browser-applications and it provides a method to verify integrity.*
  The explicit syntax of XML provides the ability to define ones own features and XML can be used to define other language, since it is a meta-language (chapter 2). Secondly, content and presentation is separated, so a XML document can be interpreted in many ways. Each XML document has to follow its structure, defined in a XML Schema. Via the Schema, the data integrity can be verified. These properties contribute to the flexible use of XML via various applications.

- *GML holds promise as an important means of storing geo-information, partly because XML's XLink and XPointer provide a means for building complex and distributed geographic data sets.*
  Geography Markup Language provides an extensive collection to encode geographical features into structured, XML-based documents. Predefined GML Schema's are developed in order to incorporate the entire range of GML definitions into an application schema. However, some advocate that this complexity has a counter effect on the usability of GML.

- *GML is based on a common model of geography, which has been developed and agreed to by the vast majority of GIS vendors in the world.*
  The key concepts of GML to model the world are drawn from the OGC Abstract Specification, which describes the world in terms of geographic features. Agreements, reached by the OGC are supported by a majority of the GIS vendors, since they are participants of the consortium. Furthermore, the GML are conform the ISO standards of geographic information.

- *GML used with XML non-spatial languages will enable spatial/non-spatial integration.*
  Often, there are difficulties ensuring the data integrity between spatial and non-spatial parts of the dataset (paragraph 4.2.2). But GML combines the spatial properties of a feature with the non-spatial part by using XML in the same document; so, spatial and attribute elements are directly connected with each other, enabling data integration.

- *SVG allows for interactivity, animation and is compatible to other XML and GML standards.*
  Scalable Vector Graphics are part of the same SGML-family as XML and GML; so compatibility is ensured. Also, SVG allows zooming, panning and other interactivity, like displaying attribute information, manipulating data layers and adding features. However, SVG has not the capabilities of a professional GIS application and especially on mobile devices, the possibilities are even more limited, which are strengthened by the limited capabilities of mobile devices.

- *GML is a logical way to pack spatial data for LBS involving mobile Internet devices.*
  GML documents are well built-up in an understandable structure and order. The advantage of GML for LBS in general, is the separation of content and presentation. This advantage makes GML suitable for display on a variety of mobile devices, whether the user has a smartphone, PDA or regular phone; the XSL transformation determines the presentation. The disadvantage of a large overhead is not a big issue, when a user only requests one simple map of an area. Thus, GML would be suitable for LBS, but for professional applications, concerning large datasets, the limitations of the mobile device and the wireless connection become a big issue. Furthermore, the lack of available software is hampering the evolution of GML for mobile devices.

6.1 – Conclusion

Geography Markup Language has all the properties to become the leading format for spatial data. It will be widely supported, since GML is an OGC specification. Also, the spatial elements of a feature are directly related with the non-spatial attributes, incorporated as XML, which ensures the data integrity. The separation between structure, content and presentation provides the ability to use the same GML document for different applications for different devices, making it suitable for many spatial related services. Drawbacks are the relatively large overhead of GML and the lack of support for complex 3D constructs.

The use of GML in desktop applications and web services is still limited, but innovations concerning the use of Geography Markup Language and other OGC specifications are at hand. New software developments are being released, which enable the user to make use of geo standards in their software solutions. The use of GML and OGC standards in mobile applications and Location-based Services are even more limited. Current hard- and software capabilities are the main reason, especially in more professional GIS applications, although GML holds a great potential, since GML can be shaped for any mobile device. Some beta-releases make use of SVG and research institutes have been able to incorporate GML on to their mobile device to display geographic maps, but commercial software vendors, like ESRI and Intergraph are still making use of their own data formats, slowing down the data interoperability.

The implemented service design of the application as described in chapter 5, proves to be suitable for gathering of point based data in the field. Although the GML component could not be incorporated on the client side of the application, it does make use of GML in the server side successfully. Other employees have immediate access to the GML data, which makes data gathering much more efficient.

It seems that the GIS industry is waiting until the concept of GML with its advantages is matured and proven to be suitable for use in desktop GIS applications until the gap between mobile and desktop can be closed. It will be a matter of time before the capabilities (processing speed, wireless connection speed) of mobile devices improve and thus the way is opened for Location-based Services and mobile applications to make use of the advantages of Geography Markup Language.

6.2 – Recommendation

The wireless application proves to be suitable for inventory or object maintenance purposes, which can accelerate fieldwork in the future. It provides a good starting point for further development and maturing of fieldwork applications. The list of attribute information can be extended, but further research is necessary on the capturing of line of polygon elements. It requires a different approach of storage, whether in a database or GML file.

First, experience with Geography Markup Language has to be gained in a desktop environment, before GML can be taken into action on mobile applications and Location-based Services. Much commercial desktop software does not yet fully support GML; imagine mobile software. The release of the WFS and WMS connectors for ArcIMS by ESRI provides a good opportunity to further investigate the concept of GML. It provides the ability to share GML data via WFS services, compliant with Open GIS Consortium specifications. Furthermore, mobile hardware performances need to improve in order to benefit from wireless GIS applications, including processing speed, storage capacity and Internet bandwidth (at the moment, UTMS is being introduced).

6.3 - Discussion

The discussion is to address points of attention.

6.3.1 - Data problems

Storing data in a database has the advantage that multiple users can use the data. But gathering data by more users at the same time can cause problems to the data model. A remark has to be first, because no multi-user application functions properly without proper agreements in advance.

First scenario that could occur is overwriting each other's data. In the application, allowing only the creator of a point to update or remove the location prevents this. This is accomplished by checking the hostname of the person, who wants to alter with the username, stored in the database.

A point data model is less difficult to manage, but when lines and polygons are involved it becomes more complicated. While standing in the field, a user can request data, which is often a small extend of the entire data model. Furthermore, the requested data layers (themes) can differ. So, when more users are conducting an inventory in the field, four situations can occur:
- Different extent (no overlap), different themes
- Different extend, same themes
- Same extend (overlap), different themes
- Same extend, same themes

When different themes are involved, the requested data layers differ and no problems occur, even when themes overlap. The gathered data is stored on the server as various layers in the data model. However, conflicts can arise, when multiple users are working on the same theme, especially when the extend interferes. First of all, it should be prohibited for multiple users to receive overlapping data. This can be accomplished by locking the databases once data is in use or by allowing a maximum extend of the requested area (based on GPS position of the users).

Other difficulties arise when objects are at the boundary of adjacent pieces of land, in particular line and polygon elements. A solution would be to restrain to possibility that polygons and lines are cut into pieces if the boundary of the spatial extend overlaps the object; only entire polygons and lines should be transferred to the user. But still, this would not solve the problem if a new polygon or line is digitized by various users (e.g. road). They should be checked by hand and connected back at the office. Corresponding coordinates have to be put together as one and only one set of attributes should exist.

References

[1] Prasad, Maneesh [2000] | 'Location Based Services' | *http://www.gisdevelopment.net/technology/lbs/techlbs003pf.htm*

[2] Magon, Ajay [2001] | 'LBS, ingredients and alternatives' | Asian GPS conference 2001 | *http://www.gisdevelopment.net/technology/lbs/techlbs006pf.htm*

[3] Maguire, David [2001] | 'Mobile Geographic Services' | InfoCity | Hyderabad, India | *http://www.gisdevelopment.net/technology/mobilemapping/techmp003pf.htm*

[4] Virrantaus, Kirsi; Markkula, Jouni [2001] | 'Developing GIS supported Location-based Services' | Helsinki University of Technology | Helsinki, Finland | *http://honeybee.helsinki.fi/GIS/GIS13/jouni_markkula_2002.pdf*

[5] Sen, Summit | 2001 | 'Open standards in location based services' | Tata Infotech | Andheri, India

[6] Fallside, David | 2001 | 'XML schema part 0: primer' | W3C | *http://www.w3.org/TR/xmlschema-0*

[7] Hiraishi, Kenji [2002] | 'The summary of OGC activities' | NEC Aerospace Systems, Ltd | Yokohama, Japan | http://www.acrors.ait.ac.th/digital_asia/GMSpapers/A91_KENJI_HIRAISHI.pdf

[8] Buehler, Kurt [2002] | 'OGC and LBS overview' | Open GIS Consortium | Wayland, United States | http://www.openls.org/dvd1/ets1/OGC_LBS_Overview.pdf

[9] Cox, Simon et al | 2003 | 'Open GIS® Geography Markup Language (GML) Implementation Specification version 3.0' | OGC | Wayland, United States | *http://www.opengis.org/docs/02-023r4.pdf*

[10] Dao, Diep [2002] | 'Location-based Services: Technical and business issues' | University of New South Wales | Sydney, Australia | *http://www.gmat.unsw.edu.au/final_year_thesis/DAO/DAO4.HTML*

[11] ESRI | 2003 | Spatial Data Standards and GIS Interoperability | Redlands, United States | *http://www.esri.com/library/whitepapers/pdfs/spatial-data-standards.pdf*

[12] ETeMII | 2001 | 'ETEMII White Paper: Chapter on Interoperability' | Parma, Italy *http://wwwlmu.jrc.it/etemii/reports/whitepaper.pdf*

[13] Carlson, David | 2001 | 'Modeling XML vocabularies with UML: part 1' | www.xmlmodeling.com | Ontogonies corp | *http://xmlmodeling.com/articles/ModelingXML-Part1.pdf*

[14] Agarwal, Ekta | 2001 | 'XML: the future of the world wide web' | *http://www.ittc.ku.edu/~sgauch/767/XML.doc*

[15] Reichardt, Mark | 2001 | 'The Open GIS Geography Markup Language (GML) Specification' | OGC | Wayland, United States | *http://www.opengis.org/docs/02-069.pdf*

[16] CHM | 2002 | 'XML vs HTML: a publishing comparison' | *http://workshops.fedstats.gov/BOCXMLvHTML.pdf*

[17] Johnston, Peter | 2002 | Metadata sharing and XML | UKOLN | *http://www.ukoln.ac.uk/nof/support/help/papers/metaxml.htm*

[18] Zapthink, LLC | 2001 | 'The pros and cons of XML' | Waltham, United States | *http://www.softinnova.com/softinnova_stuff/public/ProsConsXML.pdf*

[19] Dorninger, Peter | 2002 | 'XML technologies and geodata' | Vienna University of Technology | Vienna, Austria | *http://www.ipf.tuwien.ac.at/MarsExpress/docs/corp03/article0302.pdf*

[20] de Vries, M.E.; Tijssen, T.P.M.; Stoter, J.E.; Quak, C.W.; van Oosterom, P.J.M. | 2002 | 'The second GML prototype of the new TOP10vector object model' | Delft University | Delft, the Netherlands | *www.gdmc.nl/relay/gist9.pdf*

[21] Bray, Tim et al | 2000 | eXtensible Markup Language 1.0 | W3C | *http://www.w3.org/TR/2000/REC-xml-20001006*

[22] Houser, Alan | 2000 | 'Using Style Sheets to publish XML to the web' | FrameUsers Conference | San Diego, United States | *http://www.groupwellesley.com/talks/FrameUsers2000/stylesheets.pdf*

[23] Brentjes, Thijs | 2002 | 'Oracle en GML' | Delft University | Delft, The Netherlands | *http://www.geo.tudelft.nl/gist/publicaties/2002/CaseStud/ThijsBrentjens.zip*

[24] Lake, Ron | 2001 | GML 2.0 Enabling the Geo-spatial Web | Galdos Systems Inc | Vancouver, Canada | *http://www.galdosinc.com/newsevents/newsevent-02-10-12-files/GeoWebLocalToGlobal.ppt*

[25] Cox, Simon [2001] | 'Geography Markup Language (GML) 2.0' | OGC | Wayland, United States | *http://www.opengis.org/docs/02-069.pdf*

[26] Quak, Wilco; de Vries, Marian; Tijssen, Theo | 2002 | 'GML for exchanging topographic data' | Delft University | Delft, the Netherlands | *http://agile.isegi.unl.pt/Conference/Mallorca2002/Papers/pdf/posters/p_deWries.pdf*

[27] Taladoire, Gilles [2000] | 'Geospatial data integration and visualisation using open standard' | Université de La Nouvelle Calédonie | New Caledonia, United States | *http://wwwlmu.jrc.it/Workshops/7ec-gis/papers/pdf/taladoire.pdf*

[28] Chu Chow, Juan; Murray, Don | 2001 | 'GML – Experiences from the field' | Safe Software | Surrey, Canada | *http://www.safe.com/solutions/whitepapers/pdfs/GML%20-%20Experiences%20From%20the%20Field1.pdf*

[29] de Vita, Emmanuala; Piras, Andrea; Sanna, Stefano [2002] | 'Using compact GML to deploy interactive maps on mobile devices' | VI strada Ovest | Uta, Italy | *http://www2003.org/cdrom/papers/poster/p051/p51-devita.html*

[30] de Zeeuw, C.J. et al [2001] | 'Gebruikersspecificaties TOP10-21ste eeuw' | TDN | Emmen, the Netherlands | *http://kartoweb.itc.nl/top10nl/download/pdf/prototype.pdf*

[31] Smith, Jessica; Williamson, Ian [2001] | 'Location based services - the underlying technology' | University of Melbourne, Australia | *http://www.sli.unimelb.edu.au/research/publications/IPW/4_01Smith.pdf*

[33] Snaptrack [2003] | 'Location technologies for GSM, GRPS and UMTS networks' | Qualcomm | Campbell, United States | *http://www.cdmatech.com/resources/pdf/location_tech_wp_1-03.pdf*

[34] Ravada, Siva [1999] | 'Spatial database services for location-aware applications' | Oracle | *http://www.gisdevelopment.net/technology/lbs/techlbs004pf.htm*

[35] MySQL | MySQL reference manual | *http://www.mysql.com/doc/en/index.html*

[36] Kobben, Barend [2001] | 'RIMapper, a test bed for online Risk Indicator Maps using data-driven SVG visualisation' | ITC | Enschede, the Netherlands | *http://kartoweb.itc.nl/kobben/publications/RIMapper_paper_Vienna2004.pdf*

[37] Parsons, Ed [2003] | 'The role of web services for spatial data delivery' | Ordnance Survey | Southampton, Great-Brittain | *http://www.gita.org/members_only/downloads/eparsons.pdf*

[38] Garmin International [2001] | 'Garmin e-trex Summit owner's manual and reference guide' | Garmin | Olathe, United States | *http://www.garmin.com/manuals/etrex_sum.pdf*

[40] Reichardt, Mark [2001] | 'The Open GIS Geography Markup Language (GML) specification' | Open GIS Consortium, inc | Wayland, United States | *http://www.opengis.org/docs/02-069.pdf*

[41] Yong, Lui et al [2002] | 'Research of mobile GIS application based on handheld computer' | WuHan University | WuHan, China | *http://www.isprs.org/commission4/proceedings/pdfpapers/475.pdf*

[42] Shannon, Claude; Weaver, Warron [1947] | 'The mathematical theory of communication' |

[43] Vretanos, P.A. [2002] | 'Web Feature Service Implementation Specification' | Open GIS Consortium, inc | Wayland, United States | *http://www.opengis.org/docs/02-058.pdf*

[44] Beinat, Euro; Diaz, Eduardo [2003] | 'Location services and accuracy, an analysis for field work applications' | Spinlab, Vrije Universiteit | Amsterdam, the Netherlands

[45] Wirz, David [2004] | 'OGC Simple Features (for SQL and XML/GML)' | university of Zurich, department Geography | Zurich, Swiss | *http://www.geo.unizh.ch/oai/spatialdb/folien/ogcsf.pdf*