Software Engineering Issues in Location-Based Services

Promoteur:
Prof. Esteban Zimányi

Mémoire présenté par SERGE BOUCHER en vue de l’obtention du Diplôme d’Études Approfondies en Sciences Appliquées.
Table of Contents

TABLE OF CONTENTS ..................................................................................3

ACKNOWLEDGMENTS ..................................................................................6

1. INTRODUCTION ......................................................................................8

  1.1. Research Context ............................................................................................................... ......8

  1.2. Outline of this document....................................................................................................... ...9

2. THE LBS RECIPE ..................................................................................10

  2.1. Definition and scenarios....................................................................................................... 10

  2.2. Taxonomy of Location Based-Services ............................................................................... 12

    2.2.1. Navigation ...........................................................................................................................12

    2.2.2. Emergency ...................................................................................................................... 14

    2.2.3. Information .................................................................................................................... ......14

    2.2.4. Advertising ......................................................................................................................15

    2.2.5. Tracking .............................................................................................................................16

    2.2.6. Billing ...................................................................................................................................17

  2.3. LBS Characteristics ................................................................................................. 18

    2.3.1. Explicit / Implicit............................................................................................................ ......18

    2.3.2. Push / Pull .........................................................................................................................19

    2.3.3. Location-Based / Context-Based .......................................................................................20

3. USER DEVICES .....................................................................................22

  3.1. Short history of PED ........................................................................................................ 22

    3.1.1. Smart Phones ................................................................................................................. 23

    3.1.2. Handheld Computers ..................................................................................................... 25

    3.1.3. Car Computers .................................................................................................................29

    3.1.4. Laptop Computers .............................................................................................................30

  3.2. Ubiquitous Computing ............................................................................................... 31

  3.3. Mobile user, fixed device ............................................................................................. 32

  3.4. Synthesis ...................................................................................................................... 33
4. MOBILE COMMUNICATIONS ..............................................................34

4.1. Cellular Phones ............................................................................34
  4.1.1. 2G: GSM ..................................................................................34
  4.1.2. 2.5G: GPRS .............................................................................34
  4.1.3. 3G: UMTS ..............................................................................35
  4.1.4. Outside the cells: satellite communication ...................................36

4.2. Wireless Networking ....................................................................38
  4.2.1. WLAN ....................................................................................38
  4.2.2. WMAN ...................................................................................39
  4.2.3. WWAN ...................................................................................40

4.3. Synthesis ......................................................................................40

5. GIS .....................................................................................................42

6. POSITIONING TECHNIQUES .......................................................45

6.1. Types of location data .................................................................45

6.2. Outdoor .........................................................................................46
  6.2.1. Satellite Positioning .................................................................47
  6.2.2. Cell .........................................................................................51

6.3. Indoor .........................................................................................53
  6.3.1. Infrared ....................................................................................54
  6.3.2. Radio ......................................................................................54
  6.3.3. Ultrasound .............................................................................55
  6.3.4. Indoor GPS .............................................................................55
  6.3.5. WLAN ....................................................................................56
  6.3.6. Bluetooth ...............................................................................57

6.4. Semantic Location .........................................................................57

6.5. Seamless integration .................................................................58
Acknowledgments

I would like to begin by thanking my advisor, Prof. Esteban Zimányi, for his continuing support during my research. He gave me both the freedom to approach the subject as I wanted and invaluable advice when it was needed, and for that I’m deeply grateful.

This work also owes much to M. Sabri Skhiri dit Gabouje, who gave me much help and support during somewhat hectic times.

My friend Fabien Depasse has read many preliminary versions of this text and his input allowed me to improve it substantially. For that and for much needed support during the final weeks of writing this work, I owe him.

On a more personal note, I would like to thank Michaël Majster and Micol Lazar for their continuous friendship. They’ve been with me for most of my time at the Université Libre de Bruxelles and I wouldn’t be the person I am today had our paths never crossed.

Last, but certainly not least, my parents, and my whole family, have been incredibly supportive during all my studies. I could never have started this work but for their faith in me, and so the biggest thanks must naturally go to them.
1. Introduction

1.1. Research Context

The last few years have seen tremendous technological advances in mobile telecommunication and positioning. Handheld, networked, location-aware devices are now a reality. This opens many commercial possibilities, ranging from person locators to mobile cartography and inventory tracking.

Applications like mobile chat, fleet management, infotainment services, location-aware advertising and route finding have already been deployed, and many more are under development. But those applications are mostly built in an ad hoc fashion, resulting in wasted development time and imperfect services. LBS development is still plagued by many unsolved software engineering challenges.

The main technological barriers to Location-Based Services are behind us. Handheld electronic devices now offer significant memory and computing power. Many location technologies, either network-based or client-based, are now available. The price of wireless data transfer, while still much higher than that offered by wired link, has gone down tremendously. Yet many problems subside, and it is not clear how they should be addressed in a particular LBS project.

As we'll see, Location-Based Services software is a unique field in more ways than one. To make such a service work, data from many different sources have to be aggregated, processed, then presented in a way custom-tailored to meet user requirements. Current software development techniques are simply inadequate to deal with such heterogeneous inputs and outputs.
1.2. Outline of this document

The objective of this document is to present specific difficulties that can arise in the development of Location-Based Services and to make the case for the elaboration of a software development process that addresses those issues in a holistic manner.

Chapter 2 introduces the field of location-based services and presents a taxonomy of current and future LBS. Each service class is analyzed and specific requirements are identified.

Chapter 3 is a survey of current and future portable electronic devices suitable for the deployment of location-aware services. We attempt to classify those devices, present their current limitations and explore their likely evolution.

Chapter 4 explores the rapidly evolving field of mobile data communications and tries to identify the most promising of these for the development of LBS.

Chapter 5 is a brief introduction to Geographic Information Systems and their relevancy to LBS development.

Chapter 6 is a survey of positioning systems. It begins with a characterization of location data proprieties, then goes on to a presentation of current and future positioning devices and their applicability to location-aware services.

Chapter 7 presents ontologies, semantic data processing and Semantic Web Services. Benefits of those techniques to the development of powerful location-based services are explored.

Chapter 8 concludes this document by summarizing key findings and presenting directions for future research.
2. The LBS Recipe

2.1. Definition and scenarios

A location-based service or LBS can be defined as “A service provided to the subscriber based on her current geographic location.” (WikiLBS 2005) This definition encompasses a very broad range of services. Let’s consider how such a service could be used in practice.

Picture an Italian tourist arriving in Brussels by train one late Monday afternoon. Being of a somewhat adventurous nature, he neglected to book a hotel room in advance. He steps out of the station and, as the exhilaration of finding himself in a foreign city gently wears off, he starts to worry about finding a place to rest for the night.

Removing his trusty smart phone from his pocket, he dials into his hotel finder service. He quickly gets a list of hotels in the vicinity with vacancies for the night in his preset fare band. As he reserves a room with a single key press, the service offers to send a taxi to pick him up at his current location. He confirms, then sits down and resume reading his novel, waiting for the taxi to arrive.
This scenario describes a rather straightforward use of a typical location-based service. Although such a system is within reach of current technology, its implementation poses enough technical problems that building it in a commercially viable way would be quite difficult. Still, there’s a high probability that services of this kind will be made available to businesses and consumers in the years to come. This text is about making those services as good as they can be and their implementation as efficient as possible.

As this simple scenario illustrates, a typical LBS request operates on four key elements:

- The user’s location, obviously.
- Locations of various points of interest – such as hotels.
- Information about each POI – for example, hotel vacancies and fares.

- User context and preferences. (Here the context might be “on holiday, traveling on foot” and preferences might contain his desired hotel accommodations and maximum fare.)

In the next section we’ll explore the domain of location-based services in full and derive a classification of said services. We will see that those four key elements can be found, at least implicitly, in a very broad category of mobile applications.

2.2. **Taxonomy of Location Based-Services**

It is obvious that the following classification is overlapping: tracking and navigation services have a lot in common, and all services could in the end be considered “information” services. (G.M., Panos et al. 2003)

2.2.1. **Navigation**

The most obvious form of location-based services helps the user find his way around, by giving him information about his current location. The simplest location-based service one can envision is the **positioning device**. This service has only one function: display the user’s current location on screen, presumably in geographical latitude/longitude coordinates.

To the general public the best-known positioning system is the Global Positioning System or GPS, (Roth 2004) which has been operational since 1993. As we’ll soon see, there might be cheaper and more efficient ways to get a location aware terminal than fitting a cell-phone with a GPS receiver.
However useful latitude and longitude can be to the experienced seaman, it’s not much use for the average Joe. The next step in functionality is to provide the user with a scrollable map of his surroundings, possibly annotated with various Points of Interests – say, bars and restaurants that paid the service provider for this privilege.

The next step up is the routing service, which allows the user to input a destination and gives back a detailed and continuously updated route to it. Here the notion of user context becomes important: the route to follow will be completely different depending on whether the user is in his car or on foot. (Although most routing happens out on the streets, some situations calls for an indoor routing service – helping users find their way to a specific room in a skyscraper, for example.)

Last but not least, sometimes the target of the routing can be mobile too. In that case we generally talk of a group management service.
2.2.2. Emergency

One of the biggest problems that emergency services face when they get a call for help is that of locating the caller. Whether it’s a car that broke down, an injured person or an aggression, people who call 911 are very often only vaguely aware of their surroundings. Hence the idea that their phone could automatically transmit their location to the emergency call center, allowing help to arrive that much faster.

Recognizing that such a system would in many critical cases save lives, the US Congress passed a law asking for mobile phone companies to develop a caller-locator infrastructure by 2005 (FCC 2001) Similar legislation has been passed by the European Union. In a sense, these laws probably kick-started the whole Location-Based Service interest, by providing incentives for the development of location-aware phones.

A logical extension of this service would be to implement tracking (see below) of the vehicles. This would allow the call to be directly routed to the nearest available emergency team.

2.2.3. Information

Possibly the most appealing LBS for the consumer will be Information services. This category groups all services that provide context-dependent information to the subscriber and don’t fall in any other category.

The simplest form of information LBS is called mobile yellow pages. As the name would have it, this allows the subscriber to find specific shops and other POI around a given location.

Travel services are specifically designed for tourists and include location-aware (indoor or outdoor) guided tours, information about nearby monuments and attractions, etc.

The last category, infotainment services provide information regarding local events, theater hours and programming, etc.
2.2.4. Advertising
Some of the most controversial ideas for LBS are found in the field of location-based advertising.

One of the few currently available LBS is a mobile banner pilot application deployed in Karlsruhe public transport. On a high-resolution screen inside the train, advertisements for shops and attractions are shown as the train travels. (ELBA 2002)

A more refined form of advertisement is provided by location-triggered alerts. These are messages (e.g. SMS) delivered to the mobile user as he nears an advertising shop. Messages can be tailored to the specific needs and wants of said user, whether explicitly (by asking the user to fill surveys from time to time) or implicitly (by using Data Mining techniques on user’s past buying habits.)
In these days of barely contained spam e-mails, the idea of receiving targeted advertisement on one's mobile phone can be scary. Legislation to ban spam or restrict its expansion is not working too well. However, message sending on a mobile network is ultimately controlled and regulated by the network operator. Given the capacity of unsolicited spam to alienate subscribers, it’s highly probable that the operators will make sure that no user receives advertising messages unless he has explicitly registered for the service.

The ideal business model for such a service is still an open question: how can advertising companies both compensate subscribers for the relative inconvenience of receiving advertisements, pay for the sending of said messages and still make a profit? This will depend mainly on the kind and amount of compensation subscribers will settle for – which, as of today, is anybody’s guess.

2.2.5. Tracking

In all preceding categories, the users were mobile and the POIs were stationary. In tracking services, the opposite is true: a static user is trying to keep track of mobile assets.

Some services already allow for the tracking of children and elderly people by their legal guardian. Similar services have been used to give convicts on parole a semblance of freedom without taking chances. Corporate uses include vehicle and personnel tracking. Privacy issues are critical in such applications: targets should be allowed to switch the tracking off from their end.
A less controversial but maybe even more interesting possibility is that of **product tracking** and its uses in **Supply Chain Management** and ERP solutions.

**2.2.6. Billing**

**Location-sensitive billing** allows a mobile network provider to charge a different price depending on the location of the subscriber when he makes the call. Two obvious and very different applications are as follow:

- Supplementary charges can be imposed on more expensive customers (e.g. one calling from remote areas where an expensive network infrastructure has to be built to support few calls.)

- Mobile operators can challenge the fixed phone market more effectively by matching their prices when the mobile phone is used from home.
While both propositions are technologically feasible, they have very different business model implications. The first could alienate some consumers and drive them towards competitors with flat pricing schemes. The second one doesn’t make any sense technologically – calls are just as expensive to handle whether the phone is inside the caller’s house or a block away – but it is extremely attractive commercially. Since the mobile phone market is getting saturated, one of the only ways mobile operators can expand is by replacing fixed phones, and this means implementing location-sensitive billing.

2.3. LBS Characteristics

The taxonomy described in the previous section only dealt with the type of service provided to the user. There is another, orthogonal, classification of LBS, which focuses on more technical aspects of the service.

While some of the following characteristics may be completely transparent to the end-user, they have deep consequences in the implementation of the service, and so are well-worth studying.

2.3.1. Explicit / Implicit

We choose to call a Location-Based Service explicit if it relies on user-input for its position information. Implicit LBS derive location data from a positioning device without the user's involvement.

A typical example of explicit LBS would be a route finding site such as [www.mappy.com](http://www.mappy.com) or [www.viamichelin.com](http://www.viamichelin.com). These sites ask the user for their current location and destination by means of street addresses entered in HTML forms.

E911 services are the quintessential example of implicit LBS: the user reveals his location to the emergency services by the simple act of dialing their number.
Explicit LBS have a huge advantage: they don’t require any positioning technology. This explains why the vast majority of currently operational LBS are of the explicit type. Sadly, their ergonomics are not very satisfying, and in some case may even make the service impractical – think about in-car driving directions.

Most explicit LBS would benefit from a conversion to the implicit paradigm. Those that already have a substantial user base could do so with comparatively little risk. We can confidently predict that a substantial part of the LBS development effort in the next few years will concentrate on porting explicit lbs application to implicit models and terminals.

2.3.2. Push / Pull

Information delivery services (of which LBS are a sub-class) generally come in two flavors: push and pull.

In a push service, information is only sent to the user when he specifically asks for it. The World Wide Web is the most often-quoted pull media: it’s the user’s responsibility to request information.

In a pull service, the service provider chooses what information to send and when to send it. Television and radio are push medias.

There is room for both push and pull Location-Based Services:

- Some service types impose a pull paradigm. In a mapping service, sending the map eats up a lot of bandwidth and shouldn’t be done unless the user requests it.

- Others are naturally push services: the typical case would be advertising.
Last but not least, many types of services can be best provided by using a mixture of push and pull designs: tourism information services could be mostly pull, but most users would welcome the occasional “Interesting sight nearby” push message.

It should be noted that the push service model raises many more privacy concerns than the pull version. As most potential LBS users will have had nasty experiences with email spam, they will need ample reassurances before they subscribe to a push service. Mobile operators would do well to implement a rock-solid “do not disturb” mode that prevents any push message from reaching the user. An effective categorization of such messages, allowing the creation of filters blocking all but the most important info, would be a most helpful tool.

2.3.3. Location-Based / Context-Based
For a very long time, computer scientists believed that the most difficult part of searching was to find enough results. Google has showed that, for web searching, the number of potential results is so huge that the difficulty lies in sorting the results correctly. (Spolsky 2005)

Information LBS run into a similar conundrum: the number of hotels in a town, or more generally the number of potential POIs around the user will often get too high for the user to do the sorting manually. The small screens of most LBS terminals compound this problem.

One of the most promising ways to find the most relevant results is to focus, not on the user location, but on the user context. The user context encompasses not only his current position, but also a description of his current activity, from which his needs can be inferred. For example, let’s consider the simple request “Find me an hotel with vacancies for tonight.” Here are two possible contexts:
- The user is in the arrival hall of Brussels International Airport. He's on a business trip, traveling alone with a small handbag. His only scheduled activities are a meeting at 9am tomorrow near the Gare du Nord and a business lunch at 12:30. He's due back at the airport at 17:00 for the flight back home. His company will pay all travel and lodging expenses as long as they fall under well-known ceilings.

- The user is in the arrival hall of Brussels International Airport. He's on holiday, traveling with his wife and two kids and carrying luggage for a two weeks trip. They plan on having a light dinner in one of Brussels' downtown bistro. Tomorrow they'll rent a car in the morning and stuff their entire luggage inside then check out of the hotel. They'll spend the day walking around town before leaving for a drive to Bruges where they'll spend their next night.

Although the user is in the same location in both instances, it's clear that his needs are completely different. One of the challenges of LBS developers is to find a way to “guess” those needs with as little user input as possible.
3. **User devices**

Location-Based Services are, in essence, mobile services. They could never have been envisioned without the development of small, powerful and energy-efficient Portable Electronic Devices.

### 3.1. Short history of PED

Miniaturization has been an obsession of computer hardware engineers for decades. As soon as it became possible to build desktop computers, people have tried to make them mobile.

![An early transportable computer: The AppleIIc](image)

When as flat-panel displays and efficient batteries were invented, these computers became autonomous: they could be used for a short while without any physical connection to a wall outlet. As full-fledged laptop computers became as powerful as their desktop parents, some companies thought that sacrificing some usability and processing power would be worthwhile if it allowed for a computer that would fit in the palm of one’s hand. Those were at first dignified calendars, but in recent years they gained so much in functionality and sheer power that their only real drawback to their bigger brethrens comes from their lack of a full-size keyboard.
In parallel with this evolution, Graham Bell’s telephone was undergoing a similar metamorphosis. First it broke free from its connection to the house with the invention of the cell phone. Then, gradually, more and more functionality was added to it: calendars, note-taking and data communication were added, then small word-processors and project management software, until they became mini-computers in their own rights.

Today, the line between smart mobile phones and mini-computers with wireless networking is blurred. There exist an impressive range of powerful, compact device with Internet connection, and as more and more of them get fitted with positioning equipment, the number of possible client devices for Location-Based Services is huge. We will now explore those devices in more detail.

### 3.1.1. Smart Phones

The term *smartphone* was coined to describe cellular phones that did more than handle voice calls. According to (WikiSP 2005), the first smartphone was unveiled in 1992 by IBM. Today, every mobile phone sold has text-message capability, rudimentary contact management and calendar functions, so it’s not clear how many features a cell phone needs to be deemed “smart”.

To allow for LBS, a device needs only a few critical features:

- Position information (either embedded or provided by a separate device which connects to the smartphone, most probably using Bluetooth)

- The ability to compose a request message and to send it to an Internet server. Doing this cheaply requires access to a packet-switched data network, which as of 2005 means GPRS, EDGE or UMTS.

- The ability to present the response from the server to the human user.
Dozens of currently available phones fulfill those criteria to various degrees. But not all smartphones are created equal, and their differences are important to the LBS developer.

First, most smartphones are ill suited to entering text. Although the success of SMS-messaging seem to mean that people don’t mind typing text on a four-characters-per-button phone keyboard, the LBS developer targeting smart phones would do well to minimize typing as much as possible. (The better phones include a small QWERTY keyboard, which makes it easier to enter addresses and customization information, but they are nowhere near as efficient as laptop keyboards.)

Then, there’s no standard in smartphone Operating Systems: the market leader SymbianOS competes with among others PalmOS, Windows CE and Linux. Developing a client application that runs on most smartphones is a very tough undertaking. While more and more smartphones provide a Java Virtual Machine, running an interpreted language on hardware with limited processing power is not attractive performance-wise.
As most smartphones include a web browser, one solution is to implement the whole client-side in html. Alas, this doesn’t solve everything: designing information-dense widgets in html is quite hard, and when ten square centimeters is all the screen real-estate you’ve got it gets very challenging indeed.

The biggest problem by far is how different two smartphones can be. Screens range from monochrome low-resolution displays to VGA true-color. Input devices can be as awkward as the standard phone keypad or combine a full QWERTY keyboard with a touch-screen. Processing power of the higher-end smartphones rival those of laptops from two or three years ago.

There is no easy way to develop software clients for all those very different machines. The common denominator is so limited that it won’t be acceptable to owner’s of higher end devices, but developing for the higher end means closing out a good proportion of the customer base.

Despite all these technical problems, smartphones make the one most important platform for LBS services, for one overwhelming reason: in a few years, as all mobile phone makers run out of reasons for customers to upgrade, cell-phones will become so feature-rich that every single phone sold will be smart enough to accommodate Location-Based Services. This installed customer base dwarfs all other potential LBS platforms.

3.1.2. Handheld Computers
Early handheld computers were very much like electronic scientific calculators, with a full alphanumeric keyboard but without trigonometric functions or the ability to do anything useful. You could store addresses and phone numbers in them, write (very) short memos on the two-line screen, and most of them could serve as an alarm clock in a pinch.
They worked fine, until you wanted to export the data anywhere else, which required a “connection kit” costing three times more than the computer itself. This allowed you to copy the data to your PC in a weird file format that wasn’t readable by anything you’d ever heard of. Soon you threw the connection kit away and just retyped everything in Lotus Organizer, which took like five hours, three of them spent scrolling text on the absurdly small screen. Basically those computers were like a battery-powered spiral notebook, only much less usable – or shockproof. I know. I’ve still got four of them in a drawer somewhere.

Then came the Palm Pilot, with a big touch-screen, handwriting recognition and decent data import/export. It was a huge success. Soon an accessory modem came was released, allowing the handheld to connect to the Internet.
Fast-forward to 2005. The vast majority of handheld computers look very much like the original Palm Pilot, but they've gotten very powerful. A high-end model might incorporate a 200MHz RISC processor, 128MB internal memory and a memory-card slot for Gigabytes of additional storage. The screen is still touch-sensitive, but it's now VGA and full-color. Most importantly, extensive wireless connectivity is provided, including GSM/GPRS access.

In many ways, these high-end palmtop computers are the ideal client platform for LBS. They're small enough to be carried everywhere, powerful enough for most needs, and boast the best user interface you're likely to get in such a small package.

As palm computers have GSM modems incorporated, and cell phones are fitted with keyboards, it seems like the only difference between palm computers and smartphones is the form factor. Indeed, if you like hands-free telephony, you might just want to get a Bluetooth headset for your palm and do away with cell phones entirely.
From the LBS vendor point of view, developing for handheld computers is just like developing for smartphone, but more comfortable. Although different platforms are still competing – Windows Mobile, PalmOS, Linux, ... – the variations in processing power and input/output devices between different handhelds are much less than among smartphones. The common denominator is much more likely to provide a good enough service for most users.

On the other hand, anything that runs on a smartphone can be run on a handheld computer, only better. The potential market size for smartphones being much bigger than the one for handhelds, LBS vendors should try to make their service available for smartphones as well. This means building architectures that allows for client devices with different capabilities.
3.1.3. Car Computers
Car computers providing route-finding services are one of the oldest forms of LBS. As they predate cheap cell-phones, most of them use mass storage devices like cd-roms for the map information and don’t use networking at all. They generally use proprietary Operating Systems.

Modern LBS as discussed in this text generally use networking to provide enhanced services. Reading maps from an online server allows the route planner to take account of road works and traffic jams. As computers get more and more powerful and services get more and more sophisticated, custom operating systems are getting out of fashion.

An ideal car computer for LBS would be similar to a handheld computer, but slightly bigger. Without the need to fit in a shirt pocket, bigger screens and more useable keyboards could be used. Processing power and networking capabilities should be similar to high-end palm tops. A few companies (most notably portable computing veteran Psion Teklogix) are building those car computers right now for professional clients. Most of them seem to run some variant of the Windows CE/Mobile OS.
These are likely to be too expensive for most consumers. Most of them will just use handheld computers with “car-kits” – simple brackets that hold the device in an appropriate position for use while driving.

This is all good news for LBS vendors: developing for handheld computers will make their product automatically compatible with modern car computers.

3.1.4. Laptop Computers
Laptop computers are, well, just that: full-featured desktop computers shrunk to a size and weight you can carry anywhere and work on wherever you can sit down. Although they’ve been shrinking constantly since their invention, laptops are still big compared to the other devices discussed here. They’re also unlikely to get much smaller: you can’t fit a full-size keyboard in anything narrower than a 12” PowerBook, and if you can’t fit a decent keyboard you might as well go the whole way and build a pocket PC.
This comparatively big size means that you’re unlikely to find many LBS designed for laptops. Location-Based Services are most useful when the user is on the move, and he’s not going to sit down, open his briefcase and wait for the computer to boot in the middle of the sidewalk just to get directions to the nearest bus stop. Just about the only place where you might want to use a laptop while on the move is in a car, and that’s probably not very safe if you’re also driving. It’s a shame for LBS developer, though: laptops are now more powerful than many desktop computers, offer big screens and comfortable input devices, fast wireless network connectivity as well as full extension possibilities with fast interfaces like USB 2.0 and FireWire.

Although you may find laptops in niche location-based services where usability is more important than absolute mobility, most LBS will have to make do with much smaller screens and less convenient input devices.

### 3.2. Ubiquitous Computing

According to (Encarta 1999), for something to be *ubiquitous* means that it’s “*present everywhere at once, or seeming to be*”. Advocates of ubiquitous computing envision a future where we will be assisted by many tightly integrated computers in every step of our lives. In the words of ubicomp’s spiritual father, Mark Weiser, “*The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.*”

Today’s computing is, in a sense, already ubiquitous. The average cell phone certainly has enough processing power to be called a “computer”, and no one can deny that cell phones are everywhere. But as of now, human beings are forced into the computing world. There’s still tremendous work to do to make computers fade into the background.
It’s irritating that links to pages I’ve read on my desktop computer don’t show up as visited on my laptop, three feet away. And that’s just a minor nuisance. I’ve been writing this document on four different boxes. This required a countless number of file copies and merging, because those computers have no idea that all those partial files actually relate to the same work in progress. Without human intervention, merging those files in a coherent whole is completely impossible.

The design philosophy of ubiquitous computing is particularly relevant to Location-Based Services. Those services will be used by mostly non-technical people in situations where their minds won’t be focused on the computer screen (traffic jams, holidays in foreign towns, not to mention emergency situations). The success of those LBS will depend more on their ability to “fade into the background of the user’s life” than on any other factor.

3.3. Mobile user, fixed device

Although the traditional vision for location-based services involves a small, mobile computer device that follows the user wherever he goes, some LBS will probably be deployed on a network of fixed terminals. (The most obvious example would be tourism information kiosks. Smart advertising is another.)

Here the challenge shifts from serving a single, known user depending on his location to helping an unknown user at a fixed location. Mechanisms should be developed to identify the user and his needs, without compromising his privacy.
3.4. *Synthesis*

We hope that the preceding paragraphs have made one thing clear: there is no one accepted platform for LBS, and most services will have to be accessible from a wide ranging heterogeneous client base. This has far-reaching implications for the design of such services. To be successful, LBS frameworks and methodologies will need effective hardware abstraction layers to enable maximum code reuse between platforms.

It should be emphasized that the divergences between platforms are of a more fundamental nature than the usual hardware incompatibilities that arise for example between a Mac and a Windows computer. The latter can mostly be resolved with cross-platform widget libraries and compiling, because personal computers share similar processing power, network capabilities and I/O devices whatever they’re operating systems. On the other hand, PED differ in fundamental capabilities, so applications need to be optimized separately for each class of device, and user interfaces have to be custom designed according to I/O device constraints.

The vast majority of mobile web sites today are either custom built for a limited range of mobile devices or simply unusable on the less-capable terminals. Tool support for automatic user-interface customization is in the early stages of development, although many projects are promising. (Yun, Heiner et al. 2004)
4. Mobile communications

The vast majority of Location-Based Services require a reasonably cheap channel of communication between the user terminal and a central server. Some services can benefit from direct communication between users. We will now explore the various networking infrastructures those services can use.

4.1. Cellular Phones

Mobile phones have grown to unprecedented levels of popularity. Their unparalleled level of penetration makes them an ideal medium for the implementation of LBS communications.

4.1.1. 2G: GSM

The Global System for Mobile communications (GSM 2005) is the most widely accepted mobile phone standard today. It’s considered a second-generation system because it uses digital communications both for the signaling and speech channels.

Although it can handle data communications, GSM only supports circuit switching: it requires the establishment of a direct link between two terminals before any data can be exchanged. This means that any LBS data channel would effectively tie the phone down for as long as the service is active. This would prove both unacceptably inconvenient and unbearably expensive.

4.1.2. 2.5G: GPRS

In light of those limitations, efforts have been made to add a packet-switched networking capability to GSM networks. This allows for always-on operation without tying down much network resources, as well as a per-Megabyte pricing scheme, making data communications for LBS affordable.
In addition, GSM allows specifying different quality-of-service constraints, at varying cost. This allows for applications requiring tight transmission time constraints to get them, and for more lenient services to pay less for their bandwidth.

GPRS has been specifically designed for implementation on top of existing GSM networks with minimum modifications of the base stations and mobile terminals. This is the first reason for its success (most European mobile networks now support GPRS) but it accounts for relatively poor performance. Expected data rates are 53kbits/second for receiving and 26kbits/second for sending, and even the most stringent quality-of-service profile will allow as much as 5% of 1024bits packets to be delayed by more than 7 seconds.

Implementing sophisticated LBS on GPRS will only be possible with clever optimization of network transmissions.

4.1.3. 3G: UMTS
UMTS is the international standard destined to replace GSM networks. It aims for interoperability across borders, enhanced data rates comparable with current wired broadband access and complete integration of voice and data communications. UMTS provides an IP network with differentiated Quality of Service profiles and an always-on transfer model, all for a much lower per-megabyte price than 2G technologies.

There’s a major drawback to UMTS, though: it doesn’t exist yet, or at least not here. The only country with a sizable 3G user base is Japan. There it’s a big success, and people are already talking about 3.5G and 4G. But i-mode was a big success in Japan, and experiences to introduce it in Europe weren’t nearly as successful. Factor in the enormous licensing costs induced by UMTS and you may understand why most mobile network operators are conservative in their 3G investments.
There’s little doubt that UMTS will replace GSM in time. But in most of the world, there will be a relatively long transition time before 2G systems disappear. Most operators are upgrading their networks with GPRS and EDGE, which require much less initial investment. Massive investment in UMTS technology will have to wait for those 2.5G networks to generate enough customer interest for wireless data services.

For the time being, LBS will have to be implemented on heterogeneous networks. Wireless data transfer will be mostly limited to 2G speeds, with 3G systems initially limited to dense urban areas and gradually expanding over the years.

**4.1.4. Outside the cells: satellite communication**

Some areas of the earth are outside the coverage of any cellular phone network. For those who want vocal and data communications in truly remote areas, a few companies have deployed communication satellites that offer a wireless network over the entire earth. However, those companies are still struggling to make a profit, they face fierce competition from much cheaper cell phones, and the user base is still very small.

Although it would be technologically feasible to deploy some limited form of Location-Based Service supported by a satellite network, and this subject certainly merits attention from the research community, the technological and commercial barriers are still huge:

- Deploying and maintaining a satellite communication system available all over the earth is so phenomenally expensive, and the resulting end-user cost so high, that most satellite phone companies have virtually given up marketing to consumers. Those are all covered by cell phone networks anyway – or too poor to afford mobile communications. The only market opportunities seem to be with governments and a few very specific businesses – the oil industry being a prime customer.
- Both these cost considerations and the high round-trip time inherent in satellite communications make data transfer over satellite networks appallingly slow. (Current satellite phones achieve about 10Kbps. The Inmarsat-4 fleet – operational since mid-July – claims to reach 500Kbps, but it's use of geostationary satellites makes round-trip time ill-suited to interactive applications and the size of the terminal – roughly that of a notebook pc – doesn't allow for truly mobile computing.) Although things will probably improve in the years to come, this will require some radical reengineering in data transmission protocols and it will take some time before high-speed satellite data communication is available to the public.

- Last but not least, the vast majority of LBS scenarios simply don't make much sense in areas so remote they're outside of cell phone network coverage. If POI databases for settled regions tend to be expensive, they simply don't exist for most remote locations. Emergency services can be needed in rural areas, of course. But then there's not much good in saving half a minute with automatic position discovery and transmission when help is days away.

Satellite communication is an active field of research and commercial investments, and there's little doubt that in due time there will be a wireless data network covering the whole earth. The most promising mid-term development for LBS is the seamless cell/satellite integration promised by 3G projects. However, the lack of cooperation between the various standards organizations and commercial actors is not encouraging.

1 Although satellites can provide fast internet access to remote locations at reasonably low prices, this requires a wired connection for the uplink, making them irrelevant for LBS.
Although LBS providers would do well to keep an eye on future developments in this domain, satellite communication is currently not commercially or technologically viable as a support for LBS.

### 4.2. Wireless Networking

While cellular phone operators are extending their networks to support data communications, the computer network industry is trying to get rid of cables. Although they came from a very different direction, computer scientists have created standards for wireless networks that now compete with 3G cellular networks.

#### 4.2.1. WLAN

The original goal of Wireless Local Area Networks was to allow for the networking of computers in buildings where laying down cables was neither possible nor desirable. This explains why WLAN protocols focus on easy setup of ad-hoc networks (echoing wired Ethernet) and fast data transmission over short distances.

Currently two standards families exist for short-range wireless networking: the European HiperLAN standard (EtsiHL 2005) and the 802.11 family of IEEE standards. While the former is widely considered technologically superior, the 802.11 family already has enormous market share, and it’s possible that HiperLAN will never know widespread deployment.

802.11, however, has been a huge success. Wireless “hotspots” are all over major cities, and more are being installed as we speak. Speeds vary quite a lot depending on distance to the base station, number of terminals sharing said base station and whether there’s line-of sight visibility, but in good conditions at least one MBits/second can be achieved over distances of a few dozen meters. This is more than enough for most LBS. The small size of the hotspots even helps somehow, providing crude position information without the need for any supplementary equipment.
Alas, 802.11 is far from perfect as a LBS communication channel. It was never meant for true mobility, but rather as a cable replacement. Continuity of service across cell boundaries can be provided with Mobile-IP, but the approach is inherently inefficient and support for quality of service protocols is still an active research area. More importantly, 802.11 cells are really small. Covering an entire city with them would be prohibitively expensive. An entire building, though, poses no problem.

For LBS where its shortcomings aren’t a problem – indoor services come to mind – 802.11 provides very high bandwidth at reasonable cost.

4.2.2. WMAN

There’s currently much interest in Wireless Metropolitan Area Networks. Two competing, but partially interoperable standards are IEEE 802.16 and ETSI HiperMAN.

Those networks are very similar to the corresponding WLAN, but use much more powerful cells with longer range and more bandwidth. Typical expected values are a 50 km radius, and 70MBits/second shared data rate for a single cell. Performance degradations in the absence of a direct line of sight are to be expected.

Major installations of WMAN should be announced by the end of 2005. It will probably take a few months for the price of client equipment to come down.

WMAN is a promising technology for urban LBS with high bandwidth requirements, but due to the enormous cell size, it absolutely requires a separate positioning technology.
4.2.3. WWAN
There is currently no technology providing wireless hotspots bigger than the metropolitan class. However, the data communication facilities of satellite phone networks certainly make a wide area network. As discussed above, those facilities are currently inadequate for the implementation of Location-Based Services.

4.3. Synthesis
As we’ve seen, mobile communications are faster, cheaper and higher quality today than they’ve ever been and they should continue to improve at a steady rate in the coming years. But for the LBS developer, the situation is not an easy one. Except in some rare cases (like indoor services supported by WLAN), there is no one technology that suits the application’s needs. They’re all either too slow, too expensive, lack coverage, or are simply not available yet.

The answer to this conundrum is that good LBS should support more than one networking technology and adapt to those available. The client software may choose to connect to an 802.11 hotspot if it can find one. If not, it will fall back on the UMTS network. Failing that, it will use the 2G network. The software should adapt its data transfer policy according to the available bandwidth and quality of service – or lack thereof.

If some functionality can’t be efficiently provided with the available networking resources (e.g. photographs of attractions in a tourist information service) the system should drop them and replace them with less bandwidth-intensive data. LBS are interactive services supported by small and awkward input and output devices, so responsiveness should be the first objective. Only with such a system can the service remain useful under varying network availability.
It will be some years before UMTS networks are ubiquitous enough to support most LBS. In the meantime, developers who make best use of the heterogeneous networking resources will have a significant competitive advantage.
5. GIS

Up to now we have focussed on data transmission and delivery. But we still need to solve the hard problem of storing it in a way that allows complex querying and simultaneous availability to many users. This is where GIS come into play.

There are so many different definitions of a GIS that none is really accepted as a standard. Since we don't want to make matters worse, we'll base our definition of GIS on (N.Demers 2000)

When we talk about "a GIS" in this work, what we mean is a system which deals with space-time data. It has the following subsystems:

1. A data input subsystem that collects and pre-processes spatial data from various sources. This subsystem is also largely responsible for the transformation of different types of spatial-data (e.g. from isoline symbols on a topographic map to point elevation inside the GIS).

2. A data storage and retrieval subsystem that organizes the spatial data in a manner that allows retrieval, updating and editing.

3. A data manipulation and analysis subsystem that performs tasks on the data, aggregates and disaggregates, estimates parameters and constraints and perform modeling functions.

4. A reporting subsystem that displays all or part of the database in tabular, graphic, or map form.

The first computer system matching this definition appeared during the late 1960s. It was developed as a mainframe based system. During the 1980s and 1990s, the number of UNIX workstations – and personal computer – based GIS grown consequently. There are now a lot of different GIS solutions available for all kind of platforms. The range of features available to the end-user is incredible. From visual map analysis to automated data mining and multi-criteria decision, the only limit is the imagination.
A GIS can use information from many different sources in many different forms. As long as a variable is linked to a spatial location, it can be fed into a GIS. A lot of them even convert data which may not fit to their input format into forms they can recognize and use. The majority of digital data currently comes from photo interpretation of aerial photographs. There are two main methods to store data in a GIS: Raster and Vector.

“Raster data types consist of rows and columns of cells where in each cell is stored a single value. Most often, raster data are images (raster images), but besides just color, the value recorded for each cell may be a discrete value, such as land use, a continuous value, such as rainfall, or a null value if no data is available. [...] The resolution of the raster dataset is its cell width in ground units. For example, one cell of a raster image represents one meter on the ground. Usually cells represent square areas of the ground, but other shapes can also be used.

Vector data type uses geometries such as points, lines (series of point coordinates), or polygons, also called areas (shapes bounded by lines), to represent objects. Examples include property boundaries for a housing subdivision represented as polygons and well locations represented as points. Vector features can be made to respect spatial integrity through the application of topology rules such as 'polygons must not overlap’.” (WikiGIS 2005)
Let’s get back to our Italian tourist. When he asks his LBS terminal which are the restaurants serving pizzas in a ten miles radius around his hotel, the GIS will be able to make the link between some databases – hotel database, restaurant database, road database in this particular case, assuming they already had been pre-processed by the GIS to fit known formats and they contain the required information – and compute the answer. It will even be able to find a decent way to go from the hotel to the restaurant, draw a map of the town and highlight it. A lot of the required features are already available in systems like ArcGIS.

For all their qualities, GIS aren’t the answer to all LBS problems. They don’t solve the most difficult LBS data problem: how to process queries on heterogeneous data models with poor semantic metadata, in an efficient and highly-scalable way.

To effectively leverage GIS systems for use in mobile services, many issues need to be considered, among them: efficient data transmission and caching strategies, intelligent delegation of computation to the mobile device according to its processing capabilities, use of various levels of detail and standards compliance. (Virrantaus, Markkula et al. 2001)

GIS are generally used by domain experts to solve hard problems in a mono-user environment using complex user interfaces. LBS must be designed to serve many users using awkward terminals in an efficient way, and that’s another problem entirely.
6. Positioning techniques

After exploring the candidate physical terminals and communication technologies for Location-Based Services, we visit the last important part of the LBS recipe: positioning techniques.

We will begin by presenting a taxonomy of location information, then introduce the more relevant technologies currently available or in development. This section is mostly based on (Roth 2004) and (Giaglis, Lekakos et al. 2003), to which we refer the reader looking for a longer treatment of this subject.

6.1. Types of location data

Positioning data may come in many different flavors depending on the technology used to obtain them. This makes some of those technologies ill suited for some types of location-based services. (Roth 2004) identifies those properties for location data:

- Coordinate system: Most outdoor positioning systems provide positions in latitude-longitude coordinates. Indoor location is usually given in Cartesian coordinates relative to the building. Some technologies, like semantic locations, don’t even need the concept of a coordinate system.

- Scope: This refers to the geographical domain of positions delivered by a positioning system. Indoor positioning only works inside the building. GSM cell-id technologies are not available at sea. And satellite positioning are typically accessible over the whole earth.

- Coverage: Sometimes the system doesn’t work on all its scope. An indoor IR positioning system will have blind spots, and GSM signals are often too weak in deep valleys.
- Precision: Any measurement system suffers from inaccuracy. This can be more or less severe and is sometimes quite dependent on weather conditions and user situation. Sometimes, sources of inaccuracy can be traced and predicted (e.g. cell-positioning accuracy is directly linked to the density of base stations) and the software can exploit this to provide better service.

- Geographic vs. semantic locations: Most of the time the service doesn't really care about position coordinates; what’s really needed is information about the user’s environment. Hence positioning systems which provide information like “Bruxelles, ULB, U Building, 4th floor, B block” will often be much more useful than the geographic coordinates. (See below for a longer treatment.)

- Orientation: Sometimes (e.g. for navigation purposes) it’s valuable to know what the user’s orientation is. Most positioning systems use the difference between successive positions to compute the user’s velocity vector and deduce its orientation. But a device which measures orientation directly (like an inertial platform(WikiIP 2005)) is often more precise, and it works even in the absence of motion.

- We will now explore current positioning technologies in more detail.

### 6.2. Outdoor

Most LBS applications take place outside in the open. There are two main positioning techniques available there: satellite and cell-id. While the first one is the more precise and works anywhere on the whole globe, the latter is cheaper to implement.
6.2.1. Satellite Positioning

a) GPS

The best-known satellite positioning system, and the first to be known to the general public, is the Global Positioning System built by the US Department of Defense. It uses 24 satellites circling the earth in low-orbit, providing positioning information over the whole earth for users with small receivers. Recent models are nearly as small and cheap as cell-phones.

[Figure 6-1: Garmin eTrex Summit Handheld GPS]

Computing position is not instantaneous. On powering on, the receiver needs to acquire satellite signals, download an almanac (describing current satellite positions and orbit information) and compute the position, which requires slightly involved mathematics. First fix upon powering on can take as much as a minute, then the position gets updated once per second.
Position errors come from many factors: clocks are not perfect, satellites get slightly out of orbit due to gravitational forces from the sun and moon, and the signal from the satellites gets disrupted by the atmosphere and ionosphere. More significantly, consumer GPS don’t have access to the full system precision, which is reserved for the US army. Still, the system we get has an horizontal precision of 25 m 95% of the time, which is fine for the vast majority of applications.

Once reserved to sailors and wealthy gadget freaks, handheld GPS are being increasingly marketed to the general public. We’re seeing more and more efforts in integrating GPS with handheld computers. The Garmin iQue M5 runs Windows Mobile 2003 and is not much more expensive than a comparable non-GPS handheld.

![Garmin iQue M3 GPS/PocketPC](image)

**b) DGPS/WAAS**

For some applications GPS is not precise enough. There’s been a lot of work in improving the system, and two standards are currently commercially available.
Differential GPS uses base stations with known positions and GPS receptors. They compare their GPS position continuously, and by comparing them to their known positions compute correction values and broadcast them to nearby compatible (D)GPS receivers. (Actually, just sending the position differential wouldn’t work, so the system uses a trick called pseudo-ranges. See (Roth 2004) for the mathematical treatment.)

The Wide Area Augmentation System (WAAS) works exactly in the same way, but correction data is sent through an Inmarsat-3 geostationary satellite. Base stations are being built as we speak, and the system already covers most of North America. A similar system is being developed in Europe under the name of European Geostationary Navigation Overlay System. It should be available in early 2006 and offer positioning accuracy between 1 and 2 meters. Japan is also building a similar system under the name Multi-Functional Satellite Augmentation System (MFSAS).

Of course, you need a special GPS receiver to benefit from these systems. Most models currently sold offer DGPS, and in time every GPS will include WAAS too.

c) GLONASS, GALILEO, Beidou

GLONASS is the Russian counterpart to the American GPS system. Launched in 1996, it suffered heavy funding problems due to the economic situation in Russia, going down to a near-useless 8 satellites in 2002. It’s now back on its tracks due to the improved Russian economy and a partnership with India. Renewed operational capability is expected by 2008.
**GALILEO** (WikiGalileo 2005) is the European version of GPS. Contrary to the latter, it will be controlled by a civilian organization, not by the military. It is being developed by the European Union and the European Space Agency, with public and private funds. Many countries outside the EU are participating: Ukraine, Israel and China have already signed cooperation agreements, and other countries are expected to join. It should provide three different services: one free, with a precision of <4 meters; one reserved for government agencies; and one paying subscription service, with sub-meter precision. The system’s been designed to seamlessly integrate with existing satellite systems, and future GPS receivers will probably take advantage of both GPS and GALILEO. The first satellites should be launched by the end of 2005, with full operational capability expected by 2008.

**Beidou** is the People’s Republic of China’s independent satellite positioning technology. Instead of a constellation of low-orbit satellites, it uses a combination of four geostationary emitters (two active and two backups) and base stations to provide regional positioning reportedly as precise as DGPS. The system is currently operational over most of East Asia.

**d) Assisted GPS**

A recent development in GPS technology is that of AGPS or Assisted-GPS (Djuknic and E.Richton 2001). Specially designed for mobile phones needing position information, it relies on a simplified GPS receiver and help from the network infrastructure. Base stations with connections to mobile terminals and to the cell-phone network are used to help find the user’s position.
On powering-on, the handset gets almanac information from the AGPS server, greatly reducing the time to first fix. Then, it forwards data from GPS satellites to the server, which uses its superior computing power to compute the user’s position. The base stations are equipped with conventional GPS receivers, and since their positions are known, precision can be improved using DGPS techniques. Accuracy of 15 meters or less is expected.

AGPS trades computing power in the mobile for network usage. Given current technology, this might not be a good idea. Its saving grace might be its ability to provide rudimentary positioning indoor, with 50 meters accuracy.

e) S-UMTS
A new technology proposed in (Zeimpekis and Alvarez 2002) allows for the positioning of device using only two satellites from a typical UMTS constellation. It uses a combination of three radiolocation methods: time delay, Doppler shift and Doppler rate. According to the authors, the only drawback is an accuracy of 100-250m, making it unsuitable for the more demanding LBS.

6.2.2. Cell
Satellite positioning is a mature technology that keeps getting better, and receivers are getting cheaper and cheaper. Still, they’ll always be more expensive than a system that doesn’t require any specialized equipment. Many technologies have been developed to provide positioning information from the communication infrastructure alone, with minimal modification of the handset and the base stations.
a) Cell of Origin
The most basic positioning method is **Cell-of-Origin** or COO. Mobile networks can locate terminals by knowing which cell they currently transmit from. This technology has effectively zero deployment cost since it already exists, but it’s not very precise: in dense urban environments 200 meters can be achieved, but in rural areas cells can be kilometers large. Still, future communications technologies like GPRS and UMTS will likely use smaller cells, improving COO accuracy.

b) Timing methods
Improving on the COO technique is the **Time of arrival** technique. Base stations are modified so that the time it takes for radio waves to make a round-trip between them and the receiver can be measured. This trivially gives the distance to the base station. Repeating with two other base stations, the receiver’s position is given by the intersection of three circles.

Another method is the **Differential Time of Arrival** or **Hyperbolic** technique. This requires base stations to have synchronized clocks and constantly broadcast the current time. The receiver computes time differences between each pair of base stations, yielding hyperbolical surfaces. Their intersection gives the receiver’s position. This requires more processing power on the client side and synchronization of the base stations. But it can give an accurate fix using fewer base stations, so overall cost of the architecture is less than TOA.
c) **Directional methods**
The main method here is called **AOA or Angle of Arrival**. It relies on modification in base station allowing them to measure the heading towards the receiver. This places the user on a half-line. Combining this with a measurement of the signal strength can give a reasonable position estimate, but the better solution is to combine 2 or 3 angle measurements and compute their intersection.

d) **WLAN**
Previous paragraphs assume the terminal to locate is a cell-phone. If, on the other hand, it’s a PocketPC or a laptop and uses 802.11 connectivity, we can take advantage of the small cells of WLAN hotspots and use the Cell-Of-Origin method with decent precision – about 50 meters, which is pretty good in an outdoor LBS environment.

(Yen-Cheng, Yao-Jung et al. 2005) describes a way to obtain the hotspot id from a mobile’s IP address, enabling the deployment of LBS without any software on the terminal beyond a web browser.

### 6.3. Indoor

Technically, many outdoor positioning systems work indoor too. But most of the time they do so with reduced precision, and the most precise outdoor positioning technique, DGPS, doesn’t work at all without a clear view of the sky. More importantly, most indoor LBS require much greater absolute precision than their outdoor equivalent. (Think of a tourist information service: outdoors, we only need to find the museum; once the user enters it, the service should provide information on the exhibit currently viewed. Ditto for routing services.)

We will now present current technologies for indoor positioning. A deeper treatment can be found in (Giaglis, Pateli et al. 2002).
6.3.1. Infrared

One of the first indoor positioning systems to be developed is **Active Badge** (Roy, Andy et al. 1992). It requires users to carry an active infrared beacon that transmits a code at semi-regular intervals. Receivers are installed in each room and linked via Ethernet. Since IR signals typically don’t go through walls, this straightforwardly gives user location with room precision. Badges work for a year on a single battery.

In the more recent **WIPS** system, the roles are reversed: IR beacons are placed in fixed positions, and the user terminals are equipped with IR receivers, computing power and WLAN connectivity. Since those qualities are needed anyway for the provision of service to the user, there is no real drawback, and this system has better access control and identification.

6.3.2. Radio

Infrared cannot be used to locate devices more precisely than room-level because of wall reflections. Various systems have been proposed to locate users using radio signals.

One of these is the **SpotON** system is based on signal strength: the user wears a transmitter that sends signals at regular intervals. Receivers scattered across the building at known locations measure signal strength and send them to a central server, which computes the user’s position. Since radio transmission depends on many factors, the system has to rely on many base stations to get an accurate fix, and precision is limited at 3 meters.
6.3.3. Ultrasound
As we’ve learned in high school physics courses, sound travels much slower than electromagnetic signals. For positioning systems, that’s a huge advantage: a timing mechanism precise to 1µs (which is very easy to build) can achieve better than millimetric accuracy if it’s used to measure sound travel. In the same µs a light ray will travel 300m. On the other hand, sound is not suited for transmitting machine-readable data; so most ultrasound positioning systems require radio for signaling purposes.

The Active Bat system achieves an accuracy of 10cm. It requires the user to wear a “Bat”, which sends a short ultrasound impulse upon request from a server. Since the server only requests one bat to transmit at any given time, there is no collision danger. Receivers are arrayed on the ceiling in a 1.2m grid and connected to the server by a wired network. The server then computes the user’s position, solving equations similar to the ones used in satellite positioning. 2

In the Cricket system, transmitter and receivers are reversed. Badges are only receivers, and fixed beacons regularly transmit simultaneous ultrasound and radio signals. The time difference between their receptions gives the distance to them. Given the position of the beacons, mobile users can compute their position without a server. This makes for a more scalable system with less privacy concerns.

6.3.4. Indoor GPS
Although traditional GPS signals generally don’t penetrate inside buildings, in some large indoor environments like factories a satellite-inspired positioning system can be used. Various pseudolites (pseudo-satellites) beacons are placed around the place and transmit time information like the (real) satellites of GPS systems.

---

2 Since radio and wired transmissions are so much faster than the ultrasound signals, the time basis is the same for every receiver. This simplifies the equations quite a lot.
The biggest advantage of indoor location service is that it is designed with the small power sources of mobile devices in mind. It combines accuracy with power-efficiency, and by mimicking the signals of GPS systems, allows for indoor/outdoor GPS receivers. (Giaglis, Pateli et al. 2002)

6.3.5. WLAN
As we’ve seen, identifying 802.11 cells is an effective way to localize users in outdoor environment. The same principle works indoor, but it can’t provide the accuracy needed by many indoor LBS. Many research projects aim to use signal strength to provide improved accuracy in wireless LAN positioning.

This is much more difficult than it seems. Experience shows that the propagation of WLAN signals in indoor environments is a complex phenomenon. Signal strength can vary without the mobile terminal moving, e.g. when doors are opened or closed, furniture is moved around or people walk near the terminals. (Dricot, Ph. et al. 2003)

The first proposition came from Microsoft Research (Bahl and Padmanabhan 1999). It requires the system to be trained: the user visits several locations inside the covered area, measures the signal strength of all available access points at the location with his terminal facing various directions, and stores the resulting data in a table. When this learning process is over, the system is able to interpolate this data and use it to infer position from signal strength. Accuracy of 2-3 meters can be achieved in good conditions, but the training period is a big drawback.

A recent paper (Kitasuka, Nakanishi et al. 2003) suggests measuring the signal strength of all clients, in addition to base stations. Although it has only been tested in simulation, the system requires no calibration and its precision increases as more users enter the area. Authors claim gains in precision of up to 58% compared to AP-only methods.
6.3.6. Bluetooth
Bluetooth has been designed as the ultimate wireless data protocol. It aims to provide comparable service to USB and supports multiple simultaneous connections, data transfer and reliable identification and privacy.

In the context of indoor LBS, its short range is a big advantage. (Lauri, Nicklas et al. 2004) uses a network of Bluetooth beacons to discover nearby devices. Once a user is located, its position is sent to a central server, which can then send location-based advertising to the terminal. Since the paper focuses on Location-Based Advertising, positioning accuracy is not explored in detail. According to (Alvin, Hong et al. 2003), the Bluetooth forum is currently in the process of developing a profile for location-based service. This would substantially improve the accuracy of Bluetooth positioning methods.

6.4. Semantic Location
For most location-based services, the important data about positioning is not the absolute, geographic position, but the “meaning” of such position. Typically, a museum service aiming to provide information and documents about the exhibit currently viewed has no use for Cartesian coordinates, unless they’re necessary to figure what the user is currently looking at.

The landmark paper (Salil 2000) describes a very elegant way to get this information without resorting to geometric coordinates. Beacons are placed near points of interest, broadcasting an XML string containing relevant semantic attributes and a URL linking to a source providing more information. The paper also explores alternatives: the user device could scan barcodes placed near POI, giving the user a unique ID that could be translated to the URL by a Semantic Location Server. As a last resort, a self-positioning device could send its position to the server, which would be responsible for resolving the geographic coordinates to a meaningful semantic location.
This system also supports better privacy: beacons broadcast indiscriminately, and the user voluntarily chooses to access the URL. This decoupling between localization and information access makes it much harder for a malicious party to track users without their consent.

6.5. Seamless integration

As the preceding paragraphs should have made clear, the number of positioning systems available to LBS is huge and still growing. Not a single one of them is suited to all services, and there is no reason to believe that such a universal system will be available before long.

This raises a big problem for LBS developers: how can we build a service that can take advantage of the best positioning techniques available, and choose the most suitable in any given situation, without the user’s involvement.

Jörg Roth presents the beginning of a solution in (Roth 2003). This paper proposes the use of a middleware layer in the mobile device and a Location Server Infrastructure. The middleware layer gets the raw position data from any positioning device and sends them to the LSI, which then provides a globally unique physical location with associated semantic information. As the LSI is entirely distributed and provisions are made for the discovery of location servers, the system is evolutive and highly scalable.

But many issues are still unanswered: implementing such a platform will require trade-offs in precision and data management (esp. caching), and different services have different needs. The system also lacks a useable distance metric, making efficient routing services difficult to implement.

This is what makes LBS so hard to generalize: even though most problems can be solved in isolation, there is still no general framework able to support a wide-class of location-aware applications.
7. **LBS and the Semantic Web**

In the preceding chapters, we’ve explored some of the difficulties that an LBS developer faces: heterogeneous data sources; huge variations in client device capabilities; positioning devices differing in output format, precision and coverage, which need to provide seamless user experience. All those problems are variations on the same theme: the difficulty to process information without knowing exactly what the information is about.

7.1. **Failures of non-semantic services**

Consider those two descriptions of available rooms in Milanese hotels:

Two queen-size beds; TV set; shower; wheelchair accessible; all communication services; restaurant serves local cuisine.

Two double beds; television; small bathroom; suitable for people with disabilities; data center in the lobby; Italian restaurant in the hotel.

Which one is suitable for a tourist who needs access to email, wants pasta for supper and is traveling with one child and his handicapped wife? Are they both suitable? How can we know? Well, we know. And you know. But we’re human beings. How would a computer know?

There’s been a huge amount of research on natural language processing, and progress is being made regularly. But the short answer is: it doesn’t work. Two different hotel finding service could be advertising the very same room, both providing a complete and accurate description of it, and no computer in the world would figure out that they’re both the same thing.
When you’re preparing for a trip from the comfort of your own house, when you can open twenty web pages at the same time and leisurely read them before making your decision, this is merely an annoyance. If the same problem crops up when you’re standing outside an airport terminal in a foreign country trying to make sense of five lines of text on a difficult to read screen, chances are you’ll just give up, switch off your expensive toy and repair to the information desk.

From a developer’s perspective, the traditional way to solve this problem is to create a unique data model for your service and develop custom wrappers for every data source you may need. This is a lot of work. And when start-ups and internet portals are being created and go bankrupt in a matter of weeks, chances are you’ll spend most of your development time creating wrappers instead of improving your service.

The elegant solution is provided by Semantic Web Services.

7.2. **SWS advantages**

The core enabling technology of semantic web services is the use of *ontologies* to describe input and output data. According to (Fensel and Hendler 2005) an ontology is: “a formal, explicit specification of a shared conceptualization. A ‘conceptualization’ refers to an abstract model of some phenomenon in the world that identifies the relevant concepts of that phenomenon. ‘Explicit’ means that the type of concepts used and the constraints on their use are explicitly defined. ‘Formal’ refers to the fact that the ontology should be machine understandable. [...] ‘Shared’ reflects the notion than an ontology captures consensual knowledge, that is, it is not restricted to some individual but accepted by a group.”
In layman’s terms, an ontology allows us to describe the meaning of data in a way that a machine can understand. It allows us to formalize such common knowledge as: “double and queen-size beds are the same size”; “lodges and hotels both provide accommodations”; “vegetarian, Indian and soul-food are three types of food”; “when you’re in Milan, local and Milanese food are the thing, and a subset of Italian cuisine”.

Semantic web services provide numerous advantages. We’ll explore three: interoperability, discovery and customization.

### 7.2.1. Interoperability
Once services start to describe input parameters in semantic terms, and provide semantic information about their output data, creating portal applications that gather information from various sources become much easier.

If all services share a common ontology, this is of course trivial. But building a “super-ontology” covering every possible term is probably impossible and may not be desirable, as it’s not possible to modify all existing content to suit this new Esperanto. Also, maintaining such a mammoth quantity of data would be near-impossible.

This is why there’s a lot of research activity concerning ontology merging. This is the process of creating semantic links between concepts defined in different ontologies. For example, (Xiaomeng, Sari et al. 2004) proposes a heuristic mapping method that allows for semi-automatic ontology mapping using ontology intension and extension.
7.2.2. Discovery
One of the paradoxes in location-based services is that they can be most useful when they’re most difficult to implement, i.e. when the user is not in his usual environment. When I’m at home, I know where I can find bus schedules, because I know the URLs for Belgian public transport web sites. But when I’m visiting a foreign country, I have no idea where to look. This is why LBS providers need to make sure that their services are properly advertised, so that any potential client will find them easily.

A recent paper is looking into ways to facilitate LBS discovery during the 2008 Olympic Games in Beijing. (Norbert, enberg et al. 2004) This proposal makes heavy use of semantic web techniques.

The authors observe that users in different situations need different services. They propose a method to formally describe several user situations, and allow service providers to bind their services to these situations. The framework updates each user’s situation according to their actions and personal preferences, and offers them corresponding services accordingly.

7.2.3. Customization
All services can be improved by offering good customization facilities. In the case of Location-Based Services, adapting to the user’s needs is especially important, because the limitations of portable electronic devices don’t allow for the displaying of more than a few results per screen. Special care should be made that those few results are custom-tailored to suit the user’s needs.

Years of software usability testing has shown that most users do not take the time to customize their applications. Providing hundreds of preference options may not hurt, but 90% of users won’t ever open the Tools/Preference dialog box. If this is true for professional desktop software, it’s even more so for mobile services.
This is why location-based services developers should aim for *implicit user preference learning*. The software should progressively learn from the user’s actions what his preferences are. This process can be quite subtle, because user preferences vary over time. Quoting from (Norbert, enberg et al. 2004):

“An end user would prefer for instance Italian or French food when he or she goes to the restaurant. This can be expressed as a simple preference. On the other hand she may prefer local food *when she travels*, and the French or Italian, as usual. The condition ‘when she travels’ is related to the situation of the individual. It is what we refer to as complex preference.”

The ability to describe the user’s choice and his context in formal semantic language is a great help. As in previous examples, semantic reasoning allows the software to make sense of the user’s actions and to *deduce* his preferences from them, allowing the service to improve over time.

### 7.3. Ontologies for automatic service generation

One striking feature of the class of location-based services is that many LBS are functionally similar. Most requests have to do with geographical search for points of interests according to those points’ semantic attributes. For example, a restaurant finder and a hotel finder are programmatically identical: their only differences lie in the type of data they search.

As we’ve seen, ontologies allow us to describe knowledge domains in machine readable form. Using those descriptions, it might be possible to develop a code generator able to generate a full-fledged LBS from the domain’s description alone. This would reduce service development to ontology building, with tremendous gains in productivity and code reuse. Although this process has not yet been applied to LBS, similar projects exist. (Eriksson, Puerta et al. 1994)
8. Conclusion

8.1. Contributions of this work

Location-Based Services are an emerging field. Like in all such fields, the techniques and processes used to develop the first working products are far from optimal, because the specific problems of the field are not yet properly understood, in their specifics or in their interactions.

Many of those problems are not unheard of: user-interfaces for low-resolution screens and awkward input devices, heterogeneous data sources, limited on-board memory and heterogeneous clients raise issues that have been analyzed in the literature, although some have not seen significant research efforts until recently.

Some problems are unique to LBS: seamless integration of different positioning technologies (e.g. GPS in the wild, cells in urban areas, beacons indoor) and User Context discovery (How can the system find current user needs from his location – and location history – alone?) come to mind.

We hope that this document has made the case for the elaboration of a novel approach to the development of location-based services. The difficulties that arise in the implementation of such services cannot be solved efficiently unless they’re all considered together.

An important issue is the interaction of LBS with Geographic Information Systems. Many GIS services are available today on the Internet, and could provide the basis for successful LBS projects. There has been some research on the development of a middleware layer making data from existing GIS available to LBS, although much work still needs to be done on optimizing data transfers and adapting user interfaces to the specific needs of mobile users.
Location-Based Services could also benefit a great deal from proper use of semantic metadata. As in all systems where many different data sources have to be merged, there is no way to extract relevant information without knowing what the data actually means. Doing this automatically requires the use of formal, machine-readable Ontologies.

It so happens that these issues have never been studied together outside isolated commercial projects. It is our firm belief that such a study has to be done: each individual problem has one or more solution, but those are certainly not orthogonal and can even be conflicting. For example, network usage can be lowered if some query processing is done client-side. But in a heterogeneous environment, some clients won’t have the processing power to handle those queries efficiently, or at all.

Such problems will be encountered in most Location-Based Services in different domains. It is clear to us that the field requires a holistic framework to address those software engineering problems from requirements gathering to architectural analysis and design.

8.2. Future Research

This document is obviously just a beginning. After identifying problems in LBS engineering, the hard work is to find a way to solve them.

As we’ve seen, partial solutions already exist. Many middleware platforms have been proposed to support location-aware services, and all of them help in the implementation of certain classes of LBS in some environments. Much work is still needed to merge and generalize those solutions to obtain a more universal platform.
There is good reason to believe that a truly universal middleware layer supporting all location-aware services cannot be developed. Any design decision incurs drawbacks, and a single solution probably can’t be made to fit all services. If that is indeed the case, we need a process to identify which solution works best in each particular case.

One important aspect that this text doesn’t address is the use of peer-to-peer networks in location-based services. This has only recently begun to receive attention from the LBS community but might prove most effective in solving scalability problems.

Mobile services absolutely need user-interface customization to provide acceptable user experience across heterogeneous devices. Although only alluded to in this text, tool support to aid in this customization is slowly improving and merit further research.

More than anything else, we believe that the functional similarities between LBS have to be exploited. Many location-based services revolve around geographical searches of points-of-interests according to their semantic attributes. There might be a way to develop a service generator that takes a formal description of the domain semantics as input and semi-automatically generates a full-fledged location-based service. The potential savings in development time are more than enough to make this an exciting prospect.
Table of Figures

Figure 2-1: The ViaMichelin Hotel Finding service ................................. 11

Figure 2-2: Graphical route following on a Garmin GPS ......................... 13

Figure 2-3: Infotainment service on a mobile device ............................... 15

Figure 2-4: Friend tracking service .......................................................... 17

Figure 3-1: An early transportable computer: The AppleIIc .................... 22

Figure 3-2: High-End Nokia Smartphone .................................................... 24

Figure 3-3: Sharp Wizard handheld computer, 1990 .............................. 26

Figure 3-4: Modern Palm Pilot ................................................................. 26

Figure 3-5: iPaq PocketPC ..................................................................... 27

Figure 3-6: Garmin Car Computer and GPS ............................................. 29

Figure 3-7: Apple iBook ......................................................................... 30

Figure 6-1: Garmin eTrex Summit Handheld GPS ................................. 47

Figure 6-2: Garmin iQue M3 GPS/PocketPC ........................................... 48
Bibliography


International Conference on Software, Telecommunications and Computer Networks

ELBA (2002). Location Based Advertising Overview.


EtsiHL (2005). HiperLAN.


WikiGalileo (2005). GALILEO.
WikiGIS (2005). GIS.
WikiLBS (2005). LBS.