OpenLS for Indoor Positioning: Strategies for Standardizing Location Based Services for Indoor Use

By

Krzysztof W. Kolodziej

B.S., Geography
Rutgers University, 2000

Submitted to the Department of Urban Studies and Planning (Urban Information Systems) and
the Department of Civil and Environmental Engineering (Information Technology)
in Partial Fulfillment of the Requirements for the Degrees of

Master in City Planning
and
Masters of Engineering in Civil and
Environmental Engineering

at the Massachusetts Institute of Technology
February 2004

Signature of Author.. ......
Department of Urban Studies and Planning
February 9, 2004

Certified by..
Joseph Ferreira, Jr. Thesis Supervisor
Professor of Urban Planning and Operations Research

Certified by...
Jerome Connor, Thesis Reader
Professor of Civil and Environmental Engineering

Accepted by...
Heidi Nepf
Chairman, Committee on Graduate Students

Accepted by......................... ....
Dennis M. Frenchman, Professor of the Practice
Chairman, Master in City Planning Committee

OpenLS for Indoor Positioning: Strategies for Standardizing Location Based Services for Indoor Use

By

Krzysztof W. Kolodziej

Submitted to the Department of Urban Studies and Planning and
the Department of Civil and Environmental Engineering on February 9, 2004
in Partial Fulfillment of the Requirements for the Degrees of Master in City Planning and
Master of Engineering

ABSTRACT

The combination of location positioning technologies such as GPS and initiatives like the US Federal Communications Commission's E911 telecommunication initiatives has generated a lot of interest in applications and services that are a function of a user's location, referred to as location-based services (LBS). However, despite GPS technology and the positioning capabilities of cellular network such as GSM, millions of square meters of indoor space are out of reach of these systems. A multitude of applications and services would also benefit from indoor (in-building) positioning and navigation.

Fortunately, over the past decade, advances in location positioning technology have made it possible to locate objects indoors (in-building). These alternative technologies are now being introduced to the market enabling many kinds of indoor LBS applications. While a start, these standalone applications are unlikely to make a large impact on the marketplace, for a number of reasons discussed in this thesis.

The argument of this thesis is that in order for indoor LBS to become widely used, there is a need for both the infrastructure investment and the "killer" application (or at least a collection of sufficiently valuable applications). Without the LBS application the market will not invest in infrastructure, and without the infrastructure, the market for valuable LBS applications and their business models will not exist. The thesis distinguishes four type of infrastructure: (1) communication, (2) positioning, (3) mapping, and (4) software (services); then it argues that indoor LBS applications will need more modularity and standardization across these infrastructures in order to reach critical mass.

The aim of this thesis is to explore the extent to which open interoperability standards can have an impact on the infrastructure needed for developing indoor LBS and on the types of applications that are likely to emerge. In particular, the thesis explores location standards dealing with the application, data, and presentation layers of the Internet stack, as well as location standards from the wireless network viewpoint. Standardization can be a significant success or failure factor for any new technology, and indoor location services are no exception. This is especially true given that the overall LBS value-chain is a heterogeneous technical and business environment.

Thesis Supervisor: Joseph Ferreira, Jr.
Title: Professor of Urban Planning and Operations Research

ACKNOWLEDGMENTS

This thesis was a huge production and a big accomplishment. I don't think I would have done it alone without all the different kinds of support from the following people.

Thanks to Joseph Ferreira, my advisor, for help in framing my mind with the right set of questions, which resulted in the proper thesis structure and flow of ideas.

Thanks to Mark Reichardt, Executive Director, Open GIS Consortium, Inc. (OGC), for his willingness and enthusiasm in supporting my thesis work as well as for making OGC staff and consultants aware of the great potential and value this thesis might have to the Consortium as members think about their future steps with regard to OpenLS.

Thanks to Kurt Buhler, CEO, OGC for giving initial tips on a good start.

Thanks to Lance McKee, OGC Consultant, for general pointers on how the thesis can be of commercial value to the industry.

Thanks to Cliff Kottman, NIMA, for his thoughts on and the list of the English prepositions (spatial and temporal) that I used as part of the event language specification and interface prototype.

Thanks to Harry Niedzwiadek, CEO, Image Matters, Inc., for all the time on the phone regarding the meaning of location, location models, navigation use cases, symbolic and geometric reasoning.

Thanks to Mike Parkin, CAD/GIS Specialist, MIT Facilities, for giving me a better idea of what MIT Facilities is about and how they view standards.

Thanks to the NEXUS team folks (Christian Becker, Martin Bauer, Daniela Nicklas, and Marcus Handte) for many email exchanges regarding their state-of-the-art work with indoor positioning and how OpenGIS/OpenLS standards do and could fit in for the indoor world.

Thanks to Bodhi Priyantha, MIT Cricket, for inviting me over to the Cricket lab and explaining the how's and what's.

Thanks to Cyril Brignone, HP Labs, for time on the phone and email exchanges regarding HP Locus.

Thanks to Pete Steggles, Ubisense. Ltd., for his insider's view of indoor positioning.

Thanks to James Beal, MNSU, for many emails regarding his DCT and indoor geometric location fingerprinting research and comparing it to my relative/symbolic way of thinking about the indoor world.

Thanks to Ron Lake, President, Galdos, for email exchanges regarding the use of GML for symbolic/geometric location modeling.

Thanks to Antonio Camara, CEO, YDreams, for email exchanges and a meeting concerning indoor location-based games, the European LBS market, etc.

Thanks to Victor Bahl and Tracey Yao of Microsoft for information on Microsoft's location determination efforts.

And, special thanks go to my family: my parents, Maria and Jacek, my brother, Tomasz.

I hope I included everybody!

CHAPTER 1

INTRODUCTION: LOCATION-BASED SERVICES FOR THE "INDOOR WORLD"

1.1 Motivation, Purpose, and Scope of Study

The combination of location positioning technologies such as GPS and initiatives like the US Federal Communications Commission's E911 telecommunication initiatives has generated a lot of interest in applications and services that are a function of a user's location, referred to as location-based services (LBS). However, despite GPS technology and the positioning capabilities of cellular network such as GSM, millions of square meters of indoor space are out of reach of these systems. A multitude of applications and services would also benefit from indoor (in-building) positioning and navigation.

Fortunately, over the past decade, advances in location positioning technology have made it possible to locate objects indoors (in-building). These alternative technologies are now being introduced to the market enabling many kinds of indoor LBS applications. While a start, these standalone applications are unlikely to make a large impact on the marketplace, for a number of reasons discussed in this thesis.

The argument of this thesis is that in order for indoor LBS to become widely used, there is a need for both the infrastructure investment and the "killer" application (or at least a collection of sufficiently valuable applications). Without the LBS application the market will not invest in infrastructure, and without the infrastructure, the market for valuable LBS applications and their business models will not exist. The thesis distinguishes four type of infrastructure: (1) communication, (2) positioning, (3) mapping, and (4) software (services); then it argues that indoor LBS applications will need more modularity and standardization across these infrastructures in order to reach critical mass.

The aim of this thesis is to explore the choices, tradeoffs, and issues associated with each infrastructure type. This is done using use cases that will help solution providers understand which technologies might be best applied to solve a particular customer's requirements and what needs to be in place first for these applications to take off. In addition, these use cases will illuminate specific needs for interoperability, which yields into the overall focus of this thesis: to explore the extent to which open interoperability standards can have an impact on the infrastructure needed for developing indoor LBS and on the types of applications that are likely to emerge. In particular, the thesis explores location standards specified by the OpenGIS Consortium, Inc. (OGC)[1] dealing with the mapping and software infrastructures (i.e., application, data, and presentation layers of the Internet stack), as well as location standards specified by the

---

[1] OpenGIS Consortium, Inc. (OGC): http://www.opengis.org

Location Interoperability Forum (LIF)[2] dealing with communication and positioning infrastructures.

From the business perspective, the argument is that in order to achieve full market potential, these application, which currently are standalone and vertically implemented, need to be integrated through service-bundling. The industry's misassumption is that "killer applications" (market niches) do not appear to be "the one," when the reality is that several niches added together can constitute a very significant market. Open interoperability standards are essential to achieve this integration and are addressed throughout the thesis.

1.2 Methodology

The research is conducted using a linear method of decomposition and analysis. The methodology used to support the arguments presented in this thesis is as follows. **Chapter 2** reviews existing indoor LBS applications and explores the use/need for such applications. In addition, the technical requirements and challenges are briefly outlined through use cases in the context of each of the infrastructure types needed to deploy indoor LBS applications.

Throughout the thesis, the potential use of open interoperability standards is presented where appropriate. **Chapter 3** focuses on how well the standardization process, specifically the LIF/OMA and OGC approaches, can serve the use cases and scenario development explored in the previous chapters.

These technical requirements and challenges are then explored in detail for each infrastructure type; **Chapter 4** explores the communication and positioning infrastructures to understand how user location is positioned and tracked and provides possible solutions to problems dealing with seamless handovers; **Chapter 5** explores the mapping infrastructure, touching on location and associated content (context) data, location models, and location modeling languages; and, **Chapter 6** explores the software (services) infrastructure to understand the architectures and components needed to run indoor LBS services. **Chapter 7** presents scenarios that show what is needed to deploy most basic indoor LBS services and how to accommodate incremental growth. Moreover, the various factors and tradeoffs that may come into play when developing indoor LBS applications are explored with respect to each infrastructure type.

In **Chapter 8**, we propose a framework (composed of five dimensions) that offers a set of tasks/viewpoints and steps for generating information to help indoor LBS service designers choose between business strategy and service portfolio options. The five dimensions are marketplace, technical, organization, and economic. Filtering stages are developed to help concentrate the analysis on the most promising applications. They culminate in the selection of a service portfolio and strategy that recognize the high-level of uncertainty surrounding demand. This concluding chapter also summarizes the conclusions from each of the previous chapters and prioritizes the open interoperability standards that need to come first before certain indoor LBS services take-off.

---

[2] LIF is now part of the Open Mobile Alliance (OMA): http://www.openmobilealliance.org

1.3 Primary Research Questions

The primary research questions explored in this thesis are:

- What kinds of infrastructure are needed to deploy indoor LBS applications?
- What is the significance of interoperability and the role of standards in deploying the four infrastructure types explored in this thesis?
- What standards are mature now and for which player/provider within the LBS value-chain?

1.4 Background: The Evolution of LBS Services

The prospect of mobile phones that can determine their location was the biggest contributor to the LBS. Location determination proliferated with the US Federal Communications Commission's E911 telecommunication initiatives[3] that started in 1997. E911 requires that wireless phone providers develop a way to locate any phone that makes a 911 emergency call. To meet the FCC requirement, positioning must be accurate to within 150 meters for 95 percent of calls with receiver-based handset solutions such as GPS, or to within 300 meters with cellular's network (i.e., GSM) based approaches (i.e., cell-id). E911 is not a specific location positioning system; however, its initiatives have created a variety of location positioning systems to determine a cellular phone's location.

Location systems developed to comply with the E911 initiatives, already support LBS services for the users (i.e., wireless carrier's subscribers). For example, a wireless telephone can use this technology to find the nearest gas station, post office, movie theater, bus, or automated teller machine. To comply with E911, vendors are exploring several RF techniques, including antenna proximity, angulation using phased antenna arrays, lateration via signal attenuation and time of flight, as well as GPS-enabled handsets that transmit their computed location to the cellular system.

Some mechanisms for doing this kind of positioning do not require special hardware in the phone, but use triangulation of signals from transmission towers; a common approach is known as Enhanced Observed Time Difference (E-OTD). Another approach used by many companies is to add GPS receivers to the phone. Most of them supplement GPS data with information from a wireless network, which provides faster startup ("time to first fix") and the ability to work with much weaker signals; this is known as assisted GPS (A-GPS).

However, these established tracking technologies do not work reliably indoors. Indoor LBS requires local location positioning systems (LPSs), which use sensors to position and track users within a localized (in-building) area. These LPS systems are discussed in detail in **Chapter 4**. First, we will define what is meant by LBS.

---

[3] US Federal Communications Commission's E911 telecommunication initiatives: http://www.fcc.gov/e911

1.4.1 Definition of Location-Based Services

The term, "LBS," is a widely used term, but its definition isn't agreed upon. Kurt Buehler of the OGC provides a good general definition – "a location-based service, or LBS, is the ability to find the geographical location of a mobile device and provide services based on this location[4]." The definition's key characteristic is that LBS involves mobile users.

In principle, LBS services are:

$$\textit{Location-Based Service} = f\,(\textit{Location} + \textit{Wireless} + \textit{Internet})$$

Typically, the LBS industry places emphasis on the communication infrastructure (i.e., wireless technologies such as wireless IP platforms), and the positioning infrastructure (both explored in **Chapter 4**). While both the communication and positioning infrastructure to deliver services are undoubtedly needed, the mapping infrastructure, i.e., location models, modeling languages, application content, data format) (**Chapter 5**), and the software infrastructure (i.e., services that deliver information, Internet technologies like XML, etc) (**Chapter 6**), is what really attracts customers to buy into LBS services. This is because the full value of knowing position information will not be realized until it is applied to the contextual location of that position and the wide variety of location services that will exploit this capability (see the LBS "value-chain" in **Chapter 2**).

Overall, there are three types of LBS services, which we go into detail in **Chapter 6**:

- Simple or basic – users manually enter their location (i.e., address, phone number, place)
- Location-aware – location determined automatically (i.e., 'triggered')
- Context-aware (or ubiquitous) – adaptivity to user's activities and events

An example of an indoor LBS service in its most basic form is having the user enter his/her location manually or have the service designed in a way that it aggregates the user position to, for example, a shopping mall that the user is located in.

The second type, a location-aware service, does not require the manual position entry or aggregation. Here, adaptation mechanism are present to eliminate dependence on the slow process of users inputting information into their mobile devices, which is not user-friendly and, in turn, is one of the reasons for the slow market demand for LBS services. For example, consider a shopper in a store asking for information about the product in front of him/her. If the application is aware of the user's location, a lot of typing by the user can be avoided. Adaptivity is often defined as the relationship between a system and its environment such that a part of the system results in an optimal value of efficiency[5].

A more comprehensive LBS service of this type would guide a user inside a building to the specified destination (for example, a store location where a specified product is on sale). This

---

[4] OGC's OpenLS Initiative: http://www.openls.org

[5] Horvitz, E. Principles of Mixed-Initiative User Interfaces. In Proceedings of CHI'99, ACM SIGCHI Conference on Human Factors in Computing Systems, May 1999.

service would not only have to locate the user but also navigate him/her via map (or voice) output to support the user's way and ease orientation. Location-awareness is important in navigation where, as the surroundings change, so does the computing that takes place (the user would want the service to automatically track his/her location instead of having to enter the location along the way). Also, the issue of seamless communication and positioning handover is of high importance when providing a seamless or continuous service to the user.

The third type is context-aware[6] services, where the important aspects of an application's context in addition to location, include characteristics of environment variables (e.g., light, noise, bandwidth of network connection). In addition, personal profiles can be used to determine which information about the user's context is collected and organized to form a biography that can be consulted at a later time. One example is the ParcTab, which has been used to implement a memory prosthesis[7]. Also, combined with speech recognition technology, LBS services will revolutionize the capabilities and ease of use of mobile devices.

See **Chapter 2** for detailed use cases. See **Chapter 6** for a discussion about designing these services.

1.4.2 The "Indoor World" and the Meaning of Location

The indoor world includes settings such as shopping malls, airports, convention centers, campus, etc. LBS services are set to expand well beyond early GPS-based car navigation systems and cellular-network-based services, to encompass all location scales in a wide range of application settings[8]. The scale of the indoor world can be termed micro-geography where the concept of positioning and local mobility is different than the outdoor world.

Even though the infrastructure types that we present in the next section can be categorized the same for the outdoor world, there are subtle differences in each within the indoor world. We will get into the details later in the thesis, but what is significant to realize from the start is that for many indoor LBS applications, location needs to be determined down to a few meters and translated for the appropriate context e.g. "Fred is in Julie's room on the fifth floor." However, local positioning is pretty much about relative positioning and symbolic meaning of location (i.e., "in a room/not in a room"). This means that "xy coordinate" position is not required at such a micro-geography scale and is useless without the interpretation of the context of the user's position. This contextual representation is that the user is "in a room" on top of which LBS services can be referenced to pinpoint the user where is the nearest printer. The meaning of "nearest" is also in relative positioning terms and symbolic representation of where the printer is located in relation to the user's relative position. The notion of relative and absolute positioning is explored in detail in **Chapter 4**.

---

[6] Schilit, B., Adams, N., Want, R.: "*Context-Aware Computing Applications*", IEEE Workshop on Mobile Computing Systems and Applications.

[7] Lamming, M., Flynn, M. *"Forget-me-not"—Intimate Computing in Support of Human Memory*. Proceedings of FRIEND21, International Symposium on Next Generation Human Interface, Meguro Gajoen, Japan, 1994.

[8] Hightower, J. and Borriello, B, "Location Systems for ubiquitous computing", *IEEE Computer*, Aug 2001.

Relative location information poses the need for very short-range positioning favors particular technologies e.g. RF tags or Ultra Wideband. Passive RFID tags (attached to the user) as identified by an RFID reader (an active RF antenna, the size and sensitivity of which determine the radius of detection) are a pure identification technology, destined to become pervasive as they will be used for all manufactured items in everyday life[9].

Location has various interpretations dependent on the observer, scale, tolerance for uncertainty and errors, and many other factors. Location can be characterized as the following:

- A coordinate system such as the WGS84 datum for latitude / longitude
- A map or floorplan (in digital/computational form)
- An address (i.e., store inside a shopping mall)
- Symbolic meaning such as "Home," "Office," "Elsewhere"

Furthermore, location can be categorized according to a type:

1. **Space**: location is a physical space entity (e.g., room 23 on 5th floor, building 13 on the MIT campus)
2. **Area**: location is some space not physically demarcated, but virtually defined by applications (e.g., the area covered by a particular wireless access point)
3. **Point**: location is the position of a (mobile) user or an object. Usually, the shape and extension are not important, just in the user's or object's position (e.g., the location of a printer slate/schedule)

Location as position or area in space is a context of particular importance, and has received more attention in mobile computing than any other type of context. Like time, spatial location is an inherent attribute of other types of physical context, and often used implicitly for filtering out nearby observations as relevant context versus remote, and hence less relevant, observations. Location is a well understood context for which location models are available to support querying and processing in a variety of ways. For example, geometric models support location representations to which simple arithmetic can be applied, while symbolic models support the use of set theoretical expressions, such as '*being contained in*' (**Chapter 5**).

1.5 The Roles and Types of Infrastructure

This thesis explores four types of infrastructure essential to deploying indoor LBS: communication, positioning, mapping, and software (services). **Figure 1.1** depicts these infrastructure types and characterizes the most common communication infrastructure, a local area network (LAN), and the positioning infrastructure that leverages the Wi-Fi (and the existing LAN) for positioning, as explained in **Chapter 4**.

---

[9] Auto ID Center: http://www.autoidcenter.org

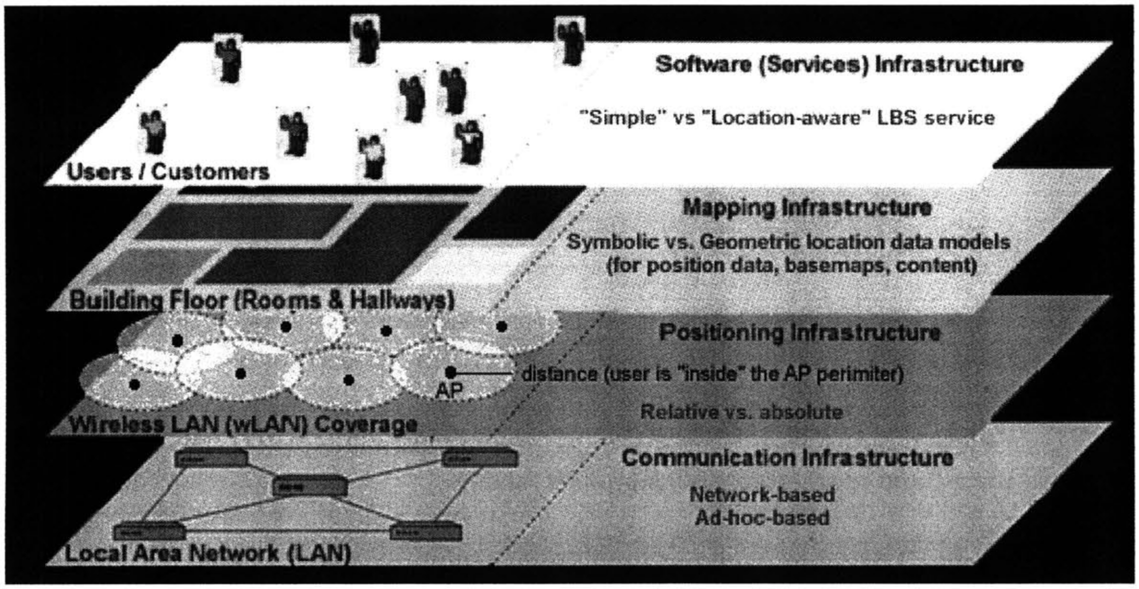
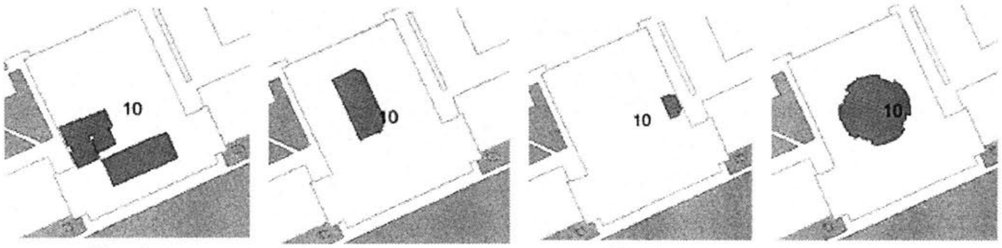


Figure 1.1: Infrastructure Types

1.5.1 The Communication Infrastructure

The communication infrastructure allows for the exchange of information among devices/computers. Obviously a cost is involved with installing any type of infrastructure. Nevertheless, in many environments of interest like shopping malls, schools, convention centers, hospital, the communication infrastructure already exists that can provide data networking capability to mobile hosts. Consider Wi-Fi networks, for example, which have been appearing everywhere; corporate campuses, college campuses, shopping malls, train stations, airports, museums, hospitals, and even Starbucks coffee shops have all installed Wi-Fi networks on their premises.

Example of an existing Wi-Fi communication network is the MIT campus where most rooms are Wi-Fi enabled. **Figure 1.2** shows a website map application used to pinpoint which portions of a floor in a particular building are Wi-Fi enabled. We will get more into this application in **Chapter 7** where we talk about MIT as a 'self-contained' company.



Floor 1 Floor 2 Floor 3 Floor 4

Figure 1.2: Wi-Fi Coverage by Room, Building 10, MIT Campus

The LBS services can complement this useful data networking capability of RF wireless LANs[10], thereby adding value to such a network. This makes a wireless LAN more valuable and can increase the chances of its large-scale deployment.

Read more about the communication infrastructure in **Chapter 4.**

1.5.2 The Positioning Infrastructure

The positioning infrastructure allows for locating the user within a space. Building a positioning system that works well indoors is a challenge, because signals reflected off walls, floors, and ceilings tend to confuse sensors, and there are often obstructions between the sensors and objects being tracked. **Figure 1.3** shows three types of sensing technologies common to the indoor world: IR and ultrasound receivers, and laser range finders.

Figure 1.3: Sensor Types (Image Source: Univ. of Washington)

The most successful research systems in this area so far have used ultrasound sensors. One example of such a system, developed at MIT, is known as Cricket[11]. Despite such successes, ultrasound-based systems require a dense network of sensors, which makes them expensive and unsuitable for most commercial applications. The most promising LPS sensing technology is Ultra Wideband (UWB). UWB can be used to implement extremely high bandwidth wireless networks and can enable users to see through walls as well as provide extremely accurate location tracking[12]. Several members of the AT&T Research team have formed a company called Ubiquitous Systems[13], which is developing LPS products based on UWB technology.

It is preferable to employ and leverage the existing communications infrastructure (i.e., Wi-Fi networks) to determine the location of users. This will decrease the deployment costs of indoor

---

[10] RF networks offer speeds of up to 11 Mbps.

[11] MIT Cricket http://nms.lcs.mit.edu/projects/cricket/

[12] UWB: http://www.uwb.org

[13] Ubiquitous Systems: http://www.ubiquitous-systems.com

LBS services. Exploiting this existing and growing communication infrastructure can provide indoor positioning comparable to what GPS is to the outdoor world.

Different indoor LBS applications have different requirements for accuracy and timing. As discussed in **Chapter 4**, determining location based on using a Wi-Fi network does not provide the best positioning accuracy, but, nevertheless, users can be positioned to room scale accuracy, which is adequate for many indoor LBS applications.

In **Chapter 4, we** provide more information about the positioning infrastructure, specific LPSs, and geo-location positioning methods.

1.5.3 The Mapping Infrastructure

The mapping infrastructure is about taking the sensor positioning information and associating it to content/context information of the world. Data location models are needed to store this information as well as modeling languages that allow for the data exchange. There are various location model types – symbolic, geometric, and hybrid – that are discussed in **Chapter 5**.

It's important to realize that different LPS systems may not express their measurements in the same coordinate frame or with the same uncertainty. There is a need for a common meaning of location, which can be achieved by a standard location model so that if multiple positioning technologies are available, their sensor information can be integrated into one model. Standard location data models as well as location modeling languages play the key role as part of our discussion of the mapping infrastructure in **Chapter 5**.

1.5.4 The Software (Service) Infrastructure

LPS technology enables a variety of indoor LBS applications. For example, by offering discounts and special deals, a shopping mall could install an LPS and persuade customers to carry tags that can be tracked. They could then offer classic LBS applications, including navigation, special offers based on location, and the ability to locate friends and family. Moreover, the mall could track extremely valuable information about customers' precise movements while shopping.

Due to the real-time nature of indoor LBS applications, the biggest challenge is the potential volume of updates for mobile users' location data, along with the requirement to analyze the updates. Traditional GIS has generally focused on more static data and long transaction updates. Trying to consolidate all location content into large warehouses poses operational issues that are costly to overcome. Typically, the most current and best content is available locally. As a result, it is better to leave location content in local, distributed warehouses (cells) and then use open interoperability standard interfaces to provide the common access mechanisms to these distributed holdings.

Most LBS applications require good integration with existing systems. LPS technology has some demanding real-time requirements, so systems tend to use distributed processing and in-memory databases for low-level tracking functionality. Depending on the situation, some applications might interact with a real-time system directly, while others might interact with an existing database management system that receives updates from the low-level tracking system.

Chapter 6 presents more about the software (service) infrastructure, software architectures, and the integration of the various components via open interoperability standard interfaces.

1.6 The Role of the Market

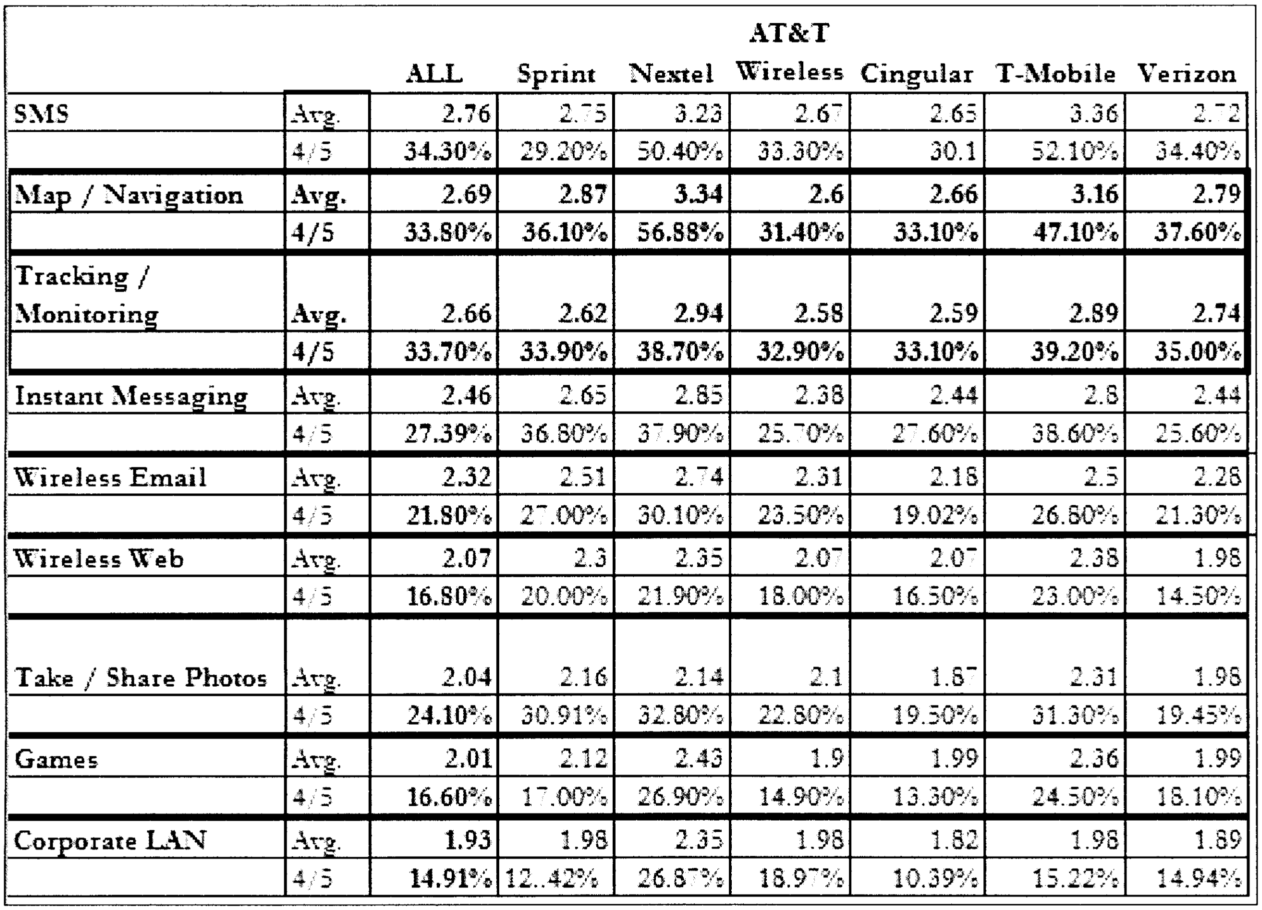
LBS technology has been touted as the next "killer app" for the Internet and wireless worlds. Every GIS company has, or is in the process of developing, a location services strategy. Most communication infrastructure providers (i.e., wireless carriers or "telecoms") have groups focused on location services. Just as is with the outdoor world where hundreds of new companies (i.e., application developers) are trying to carve a niche in the LBS industry, the indoor world will experience a similar growth. However, the LBS market faces several short-term business challenges and uncertainties that make the development and adoption of LBS application difficult.

First, just as it was with the outdoor world infrastructure providers, the indoor world might face a similar misconception or concern whether LBS can actually increase average revenue per user (ARPU)[14]. However, since the outdoor LBS market players, mainly the telecoms, have finally recognized that LBS services do drive up ARPU, this situation might not apply as much for the indoor world. This is especially true considering that telecoms are already implanting themselves with Wi-Fi networks in the indoor world, and will play a significant role both, in the indoor world as well as continue being the big market player in the outdoor world.

Market research shows that LBS is an area of unparalleled revenue potential for the operator, mainly due to the fact that subscribers are willing to pay for services that increase person- to-person communication. In fact, market research shows that LBS services ("Map/Navigation" and "Tracking/Monitoring" in **Figure 1.4**) scored higher than all other services except SMS.

---

[14] The Average Revenue Per User (ARPU) has become a magic number in the telecommunications industry. Other businesses that sell subscription do not calculate using ARPU, but because the margin of the network operator is higher the more users who use more of the operator's services, it is used in the telecommunication industry. Note that ARPU can be misleading as it is liable to both of the typical errors in calculating a mean – a small number of users can drive up the ARPU by using a service a lot, but few users make the calculation meaningless. Instead, looking at the ARPU, the median income per user and the number of users will result in a sound calculation.



		ALL	Sprint	Nextel	AT&T Wireless	Cingular	T-Mobile	Verizon
SMS	Avg.	2.76	2.75	3.23	2.67	2.65	3.36	2.72
	4/5	34.30%	29.20%	50.40%	33.30%	30.1	52.10%	34.40%
Map / Navigation	**Avg.**	**2.69**	**2.87**	**3.34**	**2.6**	**2.66**	**3.16**	**2.79**
	4/5	**33.80%**	**36.10%**	**56.88%**	**31.40%**	**33.10%**	**47.10%**	**37.60%**
Tracking / Monitoring	**Avg.**	**2.66**	**2.62**	**2.94**	**2.58**	**2.59**	**2.89**	**2.74**
	4/5	**33.70%**	**33.90%**	**38.70%**	**32.90%**	**33.10%**	**39.20%**	**35.00%**
Instant Messaging	Avg.	2.46	2.65	2.85	2.38	2.44	2.8	2.44
	4/5	27.39%	36.80%	37.90%	25.70%	27.60%	38.60%	25.60%
Wireless Email	Avg.	2.32	2.51	2.74	2.31	2.18	2.5	2.28
	4/5	21.80%	27.00%	30.10%	23.50%	19.02%	26.80%	21.30%
Wireless Web	Avg.	2.07	2.3	2.35	2.07	2.07	2.38	1.98
	4/5	16.80%	20.00%	21.90%	18.00%	16.50%	23.00%	14.50%
Take / Share Photos	Avg.	2.04	2.16	2.14	2.1	1.87	2.31	1.98
	4/5	24.10%	30.91%	32.80%	22.80%	19.50%	31.30%	19.45%
Games	Avg.	2.01	2.12	2.43	1.9	1.99	2.36	1.99
	4/5	16.60%	17.00%	26.90%	14.90%	13.30%	24.50%	18.10%
Corporate LAN	Avg.	1.93	1.98	2.35	1.98	1.82	1.98	1.89
	4/5	14.91%	12..42%	26.87%	18.97%	10.39%	15.22%	14.94%

Figure 1.4: Interest Level in Proposed Service (source: IDC)

Location-based advertising is also a source of income for the telecoms and the specific content providers (e.g.., stores in shopping malls), but it operates by means of charging the user for access to the positioning information. This extra charge is tempting for the operators, especially when they paid billions of dollars for 3G licenses they are not using to their fullest capabilities. See **Chapter 7** for the Product Finder application, where we explore the business scenario and strategies for different market players.

The most likely way of achieving revenues for LBS services (providing the application, content, etc.) is charging the user. This can be done in a number of ways: charging per location request (essentially adding a charge on top of the operator's charge for the service); charging a subscription fee (where one has to be careful to calculate the average number of location requests so that operators do not end up charging users less than what is paid to the operator); or a combination of these ways. Market research shows, however, that cellular subscribers would pay $19.63 on average for a location capability service in their new cell phones (**Figure 1.5**).

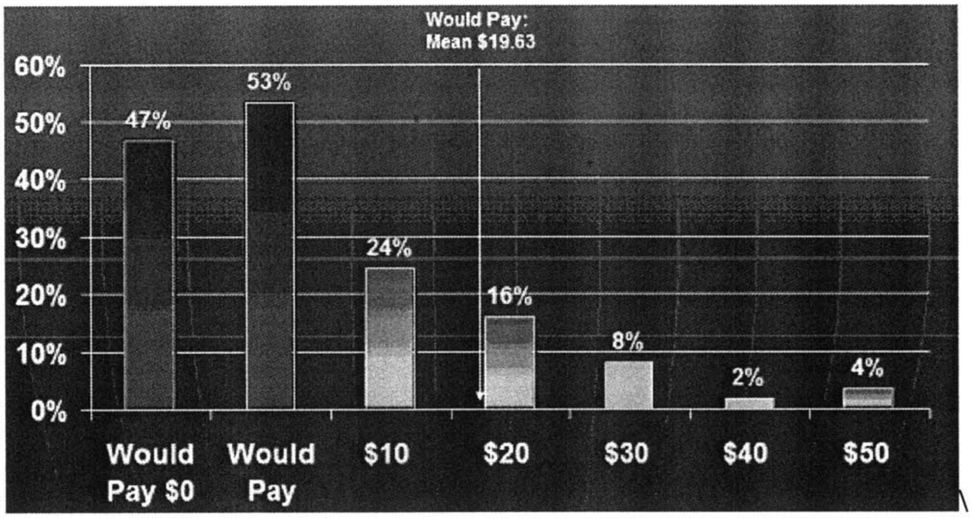


Figure 1.5: Willingness to Pay for LBS Functionality in a New Cell Phone (source: Driscoll-Wolfe[15])

Moreover, a study carried out by Driscoll-Wolfe Marketing & Research Consulting quantified customers' level of interest in, and willingness to pay for, LBS. From a survey of 20,000 households the study found that people would use routing assistance on average twice per month and look up a 'point-of-interest' less than twice per month. People with mobile devices were found to be more likely to use LBS. The research also indicates that more than 75% of respondents under age 45 would expect to use a routing assistance service at least once a month[16]. It also turns out that 16% of respondents who do not subscribe to cellular service expressed a strong level of interest in subscribing in order to obtain access to these services. An additional 15% expressed a slight interest in subscribing to cellular for location-related services. The majority (57%) of those who do not currently subscribe to cellular service would still have no interest in subscribing even if location-related services were offered at no additional cost.

Despite these figures, we believe that it will take some time for the users to adapt to these new technologies and for the market to get saturated with LBS applications such as the Buddy Finder and the Product Finder. As a matter of fact, the potential user base for consumer LBS applications is growing because most new phones are location-aware. Although consumer applications will be developed, it will be difficult for vendors to make much profit in this area. It seems likely that the telecoms will either give services away or offer them cheaply to try to differentiate their offerings.

Another challenge preventing or causing a slow growth in the LBS market is that most communication infrastructure providers have not opened up their systems/networks for the average LBS application developer. This situation is similar with the indoor LPS systems, which currently are based on closed architectures (**Chapter 6**). See **Chapter 7**, where we explore business scenario and strategies associated with closed versus open architectures. This control of

---

[15] Driscoll-Wolfe Wireless Internet Location Services Study: http://www.driscoll-wolfe.com/

[16] Driscoll C. (2001) "Are Consumers Interested in Wireless Internet Location-Based Services?" http://special.northernlight.com/wireless/w_locationbased.htm

location information is somewhat due to the subscriber's security and privacy concerns, which is discussed as part of the frameworks chapter on dimensions and filters (**Chapter 8**).

In any case, few telecoms have collaborated with the mass-market to offer location as a commodity. Some think that in general, the communication infrastructure providers would be better off leasing or wholesaling location to software (services) infrastructure provider (i.e., application developers). Instead, many telecoms have launched their own generic applications that they fully control along with their networks. Scenario 3 in **Chapter 7** talks about the open-architecture approach that was initiated and the standardization process that took place over the past years that made deploying LBS applications easier, accelerating the market penetration with LBS services.

Yet another challenge is concerned with the LBS value-chain. The adoption rates and the capability to achieve a significant percentage of communication infrastructure provider's business goals is challenged by the disparate location technology implemented by the various infrastructure types providers. Of particular concern to the outdoor world is the heterogeneous positioning infrastructures implemented by wireless carriers and service providers. Same can be said about the emerging indoor world where already there is a variety of LPSs (**Chapter 4**). See **Chapter 2** for more on the LBS value-chain.

In general, the overall LBS market, which is characterized by rapid technological change, is facing uncertainty in a range of domains:

- User value: is the customer willing to pay for these kinds of services?
- Complex networks: does the number of players involved hinder the roll-out of LBS services?
- Roles and marketing strategies: will the telecoms play the "big players" role in the indoor world as they currently do in the outdoor world? Or will non-carrier Wi-Fi network providers play a major role in niche settings such as shopping malls, airports, campuses?
- Technology: What technologies could be used and what is the level of standardization?
- Regulatory aspects: What will be the role of legislation in the development of indoor LBS?
- Public policy uncertainty: What rights do people have, concerning user position data?

Collaboration among the different infrastructure providers in the LBS market is of great importance. Take OGC's OpenLS initiative, for example, which brought together key LBS industry players to build and consolidate the standards infrastructure for outdoor LBS (**Chapter 3**). The standards and APIs developed have allowed different companies to focus on different aspects of the LBS value-chain: middleware, applications, services, etc. We speculate that a similar situation will emerge for the indoor world of LBS. This thesis hopes to be a starting point for this type of collaboration that can potentially lead into an OpenLS initiative for the indoor world.

In **Chapter 7**, we explore several scenarios that depict the potential paths along with the factors and tradeoffs that need to be considered by the infrastructure providers when deploying indoor LBS applications.

In **Chapter 8**, we outline the uncertainties surrounding the commercial provision of LBS with respect to viable business models for these services. The focus of the thesis, however, is the technological aspects of indoor LBS, specifically the infrastructure types and the role of open interoperability standards.

1.7 Thesis Structure

So far in this introductory chapter, we have briefly addressed the marketplace to indicate that LBS services are demanded by users, which makes these services profitable. In **Chapter 2**, we explore five specific use cases of indoor LBS applications that look to be promising in terms of revenues, popularity, and general interest. Here, we introduce the infrastructure requirements, which we will digest in more depth in the following chapters.

In **Chapter 3**, we introduce the standardization process and the role that open interoperability standards in deploying the infrastructure. In addition, we explain the importance of these standards in favor of enabling application integration via service-bundling that will help to achieve the full market potential (cumulative effect of combing niche markets).

In each of the chapters on the infrastructure types, we explore in detail the choices, tradeoffs, and issue associated with the components within each infrastructure. First, the communication infrastructure (**Chapter 4)** consists of network protocols (i.e., GSM, 802.11, Bluetooth) enabling interaction between the mobile device and the network on top of which LBS services are deployed. Second, the positioning infrastructure (**Chapter 4)** consists of the sensors (hardware component) and the location positioning system (software component). The interaction between the communication and the positioning infrastructure is enabled by such protocols as LIF MLP API, which is an open interoperability standard allowing for the integration of any component among these two infrastructures. Third, the mapping infrastructure (**Chapter 5)** consists of the databases that model both, the user's positions and the content/context data that is associated with the user's position for referencing which LBS services are available in that location. Fourth, and finally, the software (service) infrastructure (**Chapter 6)** consists of the different servers (i.e., application/content, GIS, navigation/routing, billing, etc.), client applications, and the middleware (i.e., Web services, location enabling connector).

In **Chapter 7**, as part of the business scenarios, we explore the business interactions among the different players in the LBS value-chain, considering that a particular player can be either a 'full-system' provider, meaning it supports all infrastructure types in deploying LBS services. Or, a 'startup' or a 'niche space' company

In the concluding chapter, **Chapter 8**, we summarize all the findings from each chapter. Using frameworks consisting of various dimensions and filters, we distill these findings by applying the most attractive pressure points and present a picture of the future for the indoor world of LBS.

CHAPTER 2

USE CASES AND A REVIEW OF EXISTING INDOOR LBS APPLICATIONS: THE REQUIREMENTS AND ISSUES

2.1 Introduction: the LBS Value-Chain

Similar to the outdoor LBS market, the indoor world of LBS is a collection of services offered by a value-chain of interconnected providers from each of the infrastructure types, such as wireless carriers, software companies, application developers, and content providers. Many of the indoor LPS systems reviewed in **Chapter 4** are self-contained systems that include all of the necessary components enabling an indoor LBS application. The relative immaturity of the indoor LBS market suggests that 'full-system' sales (including sensor hardware, data management middleware, and, perhaps, application software) will predominate for some time.

In any case, finding business cases to support individual applications can be challenging. As with so many other mobile services, it is unlikely that there will be one true killer application. Instead, the software infrastructure providers must understand how different LBS applications can be integrated to suit specific market niches as they emerge and are recognized.

The LBS value-chain (**Figure 2.1**) is where independently-provided interoperable components and the delivery of data to user on demand will lead to significant changes to the LBS marketplace. The distributed yet interoperable environment made possible by open interoperability standards has the potential to allow different companies to focus on different aspects of the LBS value-chain: systems, middleware, applications/services, data/content, etc.

Some of the outdoor LBS software (service) infrastructure providers are Autodesk, Intergraph, MapInfo, ESRI, Xmarc, Oracle, Webraska, AirFlash, Cell-loc, and CellPoint. These players may well find themselves in the indoor world of LBS in the near future, if they have not already. Furthermore, value-add content providers will probably emerge in every niche of the marketplace including, for example, shopping catalogs for the Product Finder application (these shopping catalogs already exist on the Internet, but would need to be adopted for mobile devices where screen size is of an issue). Indoor LBS players are mostly 'full system' companies (see **Chapter 4 / Appendix A)** that develop LPS systems, and include HP Locus, MIT Cricket, and MS RADAR. Other LPS that are not directly used for LBS services (but could be) include PinPoint, etc.

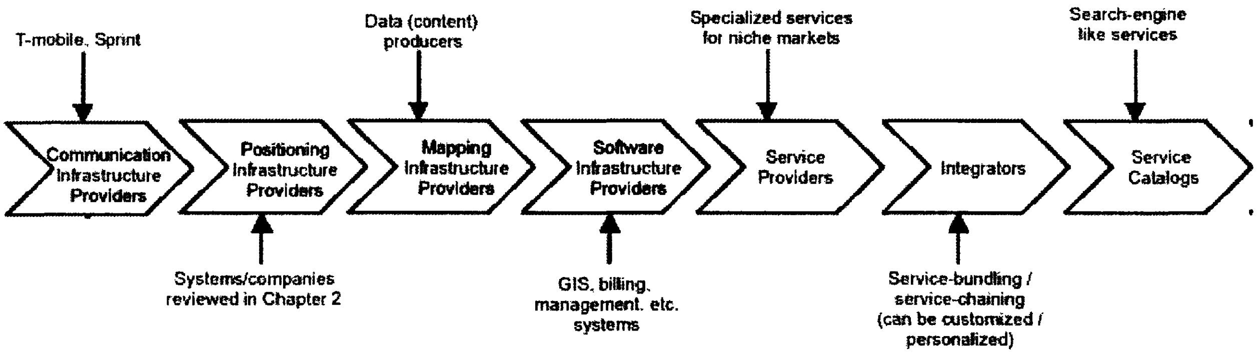


Figure 2.1: The LBS Value Chain

Before realizing the full potential of the LBS value-chain, in order for LBS applications to be successful from a business standpoint, two dimensions are essential: (1) functionality packaging and (2) niche product marketing. Indoor LBS applications such as "Buddy Finder" (launched by AT&T Wireless, Bell Mobility, Orange, and others) and "Product Finder" are emerging for the indoor world, which are featured next as part of the use cases. Overall, we think that these use cases represent what LBS developers should focus on as the "killer applications", which are economically viable, rather than frivolous services such as 'dial-your-daily-horoscope.'

2.2 Use Cases

The use cases discussed in this chapter make use of existing real-world indoor LBS applications (most of them are still "in the lab") for discussion purposes of the usage, requirements, and issues of such applications. The discussion doesn't necessarily reflect the actual functionality of the real-world application. It does, however, explore these applications in the context of the requirements and challenges for each infrastructure types (common to all of these applications), which are presented in detail in the following chapters. For the positioning infrastructure (**Chapter 4**), these requirements are associated with accuracy requirements of these applications and pertain to relative versus absolute positioning. The challenges include the "seamless" communication and positioning service continuity, where a "mutli-world" environment (i.e., outdoor to indoor, one building versus multiple buildings, one floor versus multiple floors) might potentially be composed of different communication networks and positioning technologies. For the mapping infrastructure (**Chapter 5**), requirements include modeling the user's position in relation to the rest of the "world" and pertains to geometric versus symbolic location data models. The challenges include the notion of "multi-worlds" and "seamless" content handover. And, for the software (service) infrastructure (**Chapter 6**), the requirements are associated with offering either a "simple or basic" or "location-aware" LBS service.

Moreover, the use cases briefly indicate where open interoperability standards from LIF and OGC fit in and where such standards might be limited or lacking. The role of standards will be explored in more depth within each following chapter.

The general assumption is that the user's location is known by virtue of known device location (as determined by the location positioning server/service). The service is invoked either by the user's actions on a mobile device or the application running on the mobile device is setup with, for example, preferences and options. These services are invoked through a Web browser application or from an independent location service client that operates on a mobile device.

2.2.1 Use Case 1: Point/Place of Interest (POI) (Static Objects)

> *(1) "I want to see a map of my location and what objects (static) and/or services is around me..."or (2) "I want to see map of some other location...maybe another floor of the building..."and/or (3) "I want to select a map feature to get more information ..." and/or (4) "I want to see 'virtual' information about a feature..."*

(1) Given a device location and a map/feature (or a "node" in case of symbolic models) query filter, obtain a map or feature and display it on the device. In an information service, information is filtered with respect to a specific location and the location of the user. This means providing information to the user about his/her surroundings based on preference's such as removing all stores that are not shoe stores from the map.

(2) Given any location and a map/feature query filter, obtain a map or feature and display it on the device. In a real-world application, the Vodaphone-controlled mobile operator Jphone in Japan has the functionality that enables users to get a map of their own position, and to email it to friends who have problems finding them. Although this is an outdoor LBS service, its usage potential applies to indoor spaces, too.

(3) Given a map display and a cursor/pointer, all the user to 'pinpoint' a map feature and obtain its properties. This would require the map to be either in a feature vector format (geometric map), or have links hovering on top of the features that are selectable (symbolic map). The properties might be any information that is associated with the feature, such as the service operation hours and charge fee (i.e., printing service, restaurant menu, and address/phone number of a store). For features that are places of business, the default properties might be the yellow page listing(s) at that location.

(4) Sometimes, there might some 'virtual' information about a feature such as a website that contains location-based resources.

A real world example of this sort of application is MIT Cricket's Floorplan, which is an active map utility that uses MIT's Cricket (**Chapter 4**) and a map server (**Chapter 6**) to present a location-dependent "active" map to the user, highlighting the user's location on it as he/she moves. The Floorplan loads map images from the map server, which also provides the values of (x, y) coordinate on the map corresponding to the user's current virtual space (vspace) position. As the user moves around the building, the listener (RF beckons) infers its location and asks the map server to provide the location on the map. Also, The Floorplan also displays the set of services (as icons) that are located in the vicinity of the user, which are dynamically updated as the user moves. These icons are displayed on the map; when the user clicks on an icon. The

Floorplan uses INS to download a control script or program for the application represented by that icon, and loads the controls into a new window so the user can control the application.

Figure 2.2 shows an active map displayed by the Floorplan application where the user is in room 504 (represented by the dot). It also displays four services it has found in the environment (space): an MP3 service (represented by the speaker icon) in room 504; a TV service (represented by the TV icon) in room 504, and two printers (represented by the printer icons) in room 517. Using this, a user with no knowledge of his/her environment or software to control services within it can bootstrap himself/herself with no manual configuration.

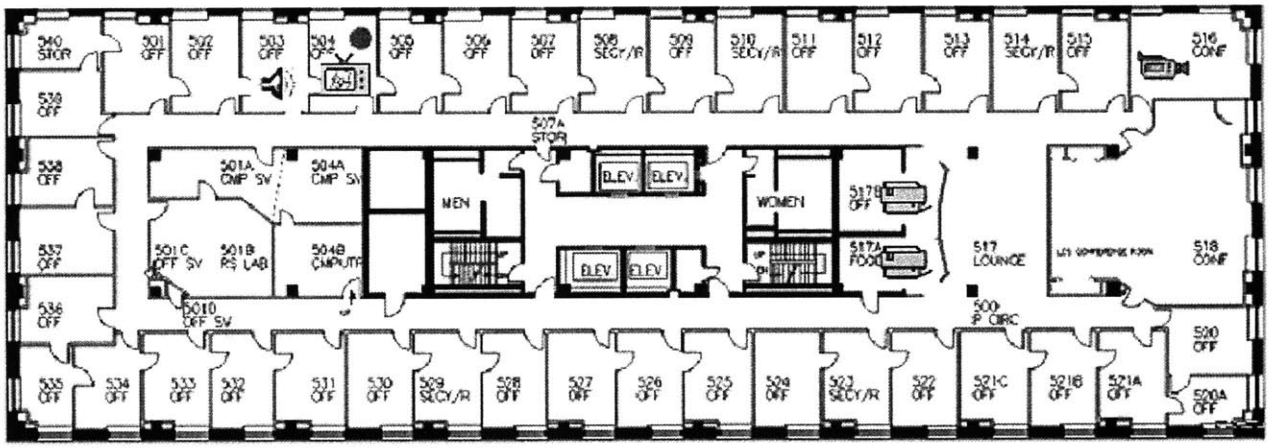


Figure 2.2: Floorplan "Active" Map (Image Source: MIT Cricket)

Infrastructure Types Characteristics Overview	
Scale:	Floor / room
Infrastructure type:	
1. Communication	
Network type	Service-based
2. Positioning	
Positioning type	Absolute
Accuracy/ Precision	Room-level (4 x 4 ft regions) / few inches
Orientation technology	MIT Cricket's compass software
3. Mapping	
Data model type	Geometric (x,y coordinates)
Modeling language	Needs a location component
4. Software (Service)	
Architecture-type	Closed-architecture (application-centric)
Service type	"Location-aware" (reactive - central server tells user where a beacon is located or gives him/her services that are within that area)

Table 2.1: Use Case 1 – Infrastructure Types Characteristics Overview

Infrastructure Requirements and Potential Use of Standards

In terms of the communication infrastructure, the main requirement is that the user stays connected while moving around the building. This is of concern especially when the building is equipped with different communication networks (i.e., GSM, Wi-Fi). The concept of seamless communication handover is explored in **Chapter 4**.

Similarly, in terms of the positioning infrastructure, the requirement is that the user is located (or tracked) in real-time at all times when needed. Location positioning can be of two types: relative (i.e., in the vicinity) or absolute (i.e., x, y coordinates). This use case is about absolute positioning since the LPS MIT Cricket gives x, y coordinate positions. However, other LPS systems are designed to represent position as relative with respect to other objects/entities. The concept of seamless positioning (location) handover is explored in **Chapter 4**. The LIF MLP API standard is explored as it applies to relative and absolute positioning.

In terms of the mapping infrastructure, there is a need to model and map geometric points such as user's locations. Location of other objects like printers, are symbolic (printer A is "in room 101"). **Chapter 5** discusses the use of geometric and symbolic location models and how the two can be integrated in order to achieve a seamless content handover.

In terms of the software infrastructure, the architecture must be scaleable to adopt to more dynamic location updates.
Also, the LBS service needs to use a query filter, preferably XML-based (such as OGC's Filter Encoding specification) to specify the type of location content a user wants to see displayed. For map retrieval and display OGC's Web Mapping Service (WMS) specification can be utilized. Using OGC's WMS request syntax, a map request can also contain a 'GetFeatureInfo' call, meaning that even though the map is in a raster image format, as it is with the FloorPlan ActiveMap application, information about a feature can be obtained that is part of the map image such as, for example, the service(s) that is marked on the map. In **Chapter 7**, as part of our scenarios, we explore why a provider should (and should not) pay attention to such open interoperability standards like OGC WMS and Filter Encoding.

2.2.2 Use Case 2: Location-Based Triggers ("Buddy Finder," "Conference Assistant" and "Child Alert")

> *"While in the shopping mall, I want to use my 'Instant Messaging Buddy Alert Service' to notify me when my friends or family are in the mall..."*

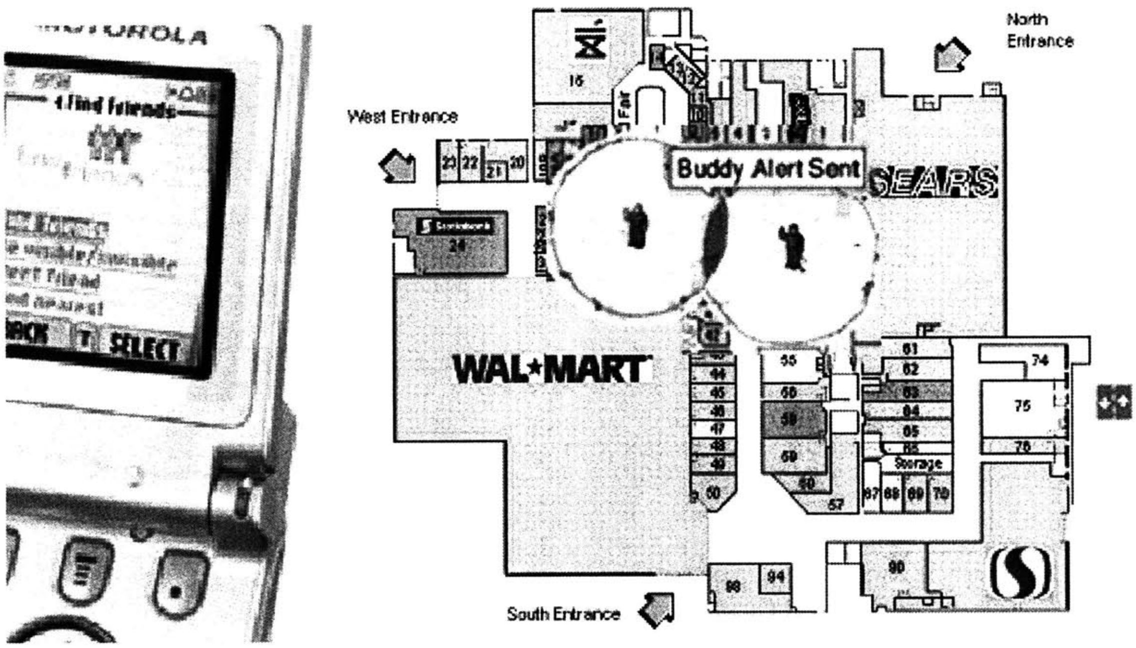


Figure 2.3: 'Buddy Finder' Alert-Based LBS Service

Indoor positioning technologies open the possibility of relating people, objects and events in space at, for example, a shopping mall or convention centers. Setting meetings in points of interest and facilitating the access to those points (i.e., schedule a meeting at a given time in a given spot and provide optimal routing for all involved). Another use is finding people with certain features in the proximity (i.e., date matching service).

Alert-based LBS, also called spatial-trigger (proactive) services, are among the most fundamental in importance. The underlying premise is to notify a user if something (an event or activity) important happens, which in this case is when a friend ('buddy') enters the specified perimeter area (or zone). Assume that *'user1'* is traveling to a shopping mall. *'User1'* wishes to consult with *'User2'* (a friend) who will also be going shopping to the same mall. *'User1'* consults his cell phone LBS "Buddy Finder' service and initiates the 'alert' notification to be triggered when *'User2'* is in the vicinity. (This service would consult *'User1''s* user profile from a user profile provider to generate a 'buddy list' that includes User2.) The 'Buddy Finder' service then uses the location information of the communication network's Wi-Fi location capability (**Chapter 4**) in both users' cell phones to determine the proximity of their owners.

This 'spatial messaging' concept is similar to the instant messenger service of ICQ, MSN IM, or AOL IM, which can tell the user whether their contacts are currently online. This is a virtual meeting place. The same can be true for a physical interaction, whether personal or business. The user is alerted when their contacts are in the vicinity. If prearranged, people can find each other quicker. Teens, for example, will want to receive an SMS that friends or a potential match for a dream date is close-by. Moreover, the proactive nature enables impromptu gatherings (ad-hoc meetings). A real-world example of this kind of service is NTT DoCoMo's Friends-Finder service in Japan. This service was developed for outdoor interaction, which reports people's location when they come within half a mile of their contacts[17]. An indoor LBS version of this service would require the perimeter range (vicinity) to be smaller to a building scale (i.e., shopping mall). Another extension to this application would be a dating service, where dates are arranged by vicinity in addition to user (date) profiling[18].

This kind of a location-aware service can provide opportunities to informally meet people through a specification language (**Appendix C**) and interface through which the user can specify one or more contexts, in this case, location, in which the user thinks such opportunities might exist. For example, if the user needs to ask another person a question, the user could ask the service to notify him/her when that person returns to some location (i.e., his/her office room). This concept is being developed by Accenture with their EventManager software[19]. Microsoft's MSN IM Buddy List service is another real-world example of location-aware services (i.e., "*Alert me when you find Harry*" or "*Alert me when Harry happens to be near by*"). Moreover, if two users are seen together in the same location, the service could trigger further action and play them a video message on their mobile device, or the nearest LCD screen display.

Another example of a triggered indoor LBS service is the "Conference Assistant[20]," which falls under the 'community building' application category. The assistant uses a variety of context information to help conference attendees. It also examines the conference schedule, topics of presentations, user's location, and user's research interests to suggest the presentations to attend. Whenever the user enters a presentation room, the assistant automatically (by a location-based trigger) displays the name of the presenter, the title of the presentation, and other related information. Available audio and video equipment automatically record the slides of current presentation, comments, and questions for later retrieval.

Yet another example of a triggered indoor LBS service is a "Child Trigger," which falls under the 'safety and security' application category. Adult users can gain comfort in knowing that their child is safe as they will be notified if their child or elderly family member has ventured beyond the security of a pre-specified "safe" region (i.e., child leaves school during teaching hours). These family oriented security services are priceless to parents and relatives as all types of time and location-sensitive information have incomparable value to users. With respect to profit making, the major consumer market driver will be information services, with rapid growth predicted for personal and child safety services[21].

---

[17] Boswell, R. Location-Based Technology Pushes the Edge. Telecommunications, June 2000.

[18] Hendrey, G. Managing the wireless internet. RF Design, March 2001, p50-56.

[19] McCarthy, J. F. and Anagnost, T. D. EventManager: Support for the Peripheral Awareness of Events. In P. Thomas and H. W. Gellersen, editors, Handheld and Ubiquitous Computing, number 1927 in Lecture Notes in Computer Sciences, page 227 to 236. Springer Verlag, Germany, September 2000.

[20] Dey, A., Futakawa, M., Salber, D., and Abowd, G. The Conference Assistant: Combining Context-Awareness with Wearable Computing. In Proceedings of the 3rd International Symposium on Wearable Computers (ISWC '99), pages 21-28, San Francisco, CA, October 1999. IEEE Computer Society Press.

[21] 3G: http://www.3g.co.uk/PR/Oct2003/5950.htm

Infrastructure Types Characteristics Overview	
Scale:	Building/floor
Infrastructure type:	
1. Communication	
Network type	Service-based
2. Positioning	
Positioning type	Relative (Cell-id)
Accuracy / Precision	Building-level / Room-level
3. Mapping	
Location model type:	Symbolic
Modeling language	Needs a location + event + time components
4. Software (Services)	
Architecture-type	Closed-architecture (application-centric)
Service type	“Location-aware” (proactive)

Table 2.2: Use Case 2 – Infrastructure Types Characteristics Overview

Infrastructure Requirements and Potential Use of Standards

Similarly to the previous use case, a seamless handover for both communication and positioning (location) is a requirement. The positioning accuracy for this use case is building-scale. Note, however, that a Buddy Finder application, for example, might also cause a location-trigger if the users (friends) are in the same room. This implies that positioning needs to be represented (modeled) on several scales to make the application/service scalable.

In terms of the mapping infrastructure, note that all of these location-trigger based applications don’t really require a map to be displayed on the mobile device. However, location still needs to be represented (modeled) somehow. Here, the location is symbolic (i.e., “inside the mall”), which is linked to positioning that is relative of nature. Also, the modeling language does not only require a location component, but also an event and time component. This requirement is seen when the user “enters” a conference rooms. The event here is that the user is walking in from one zone (hallway) to another zone (room). The time component can be used as a restriction for the Buddy Finder service, to notify the user of his/her friends that are in the vicinity only between 1pm to 4pm on Fridays, for example. The syntax for such a *location + event + temporal* language could be prototyped as the following:

When/If *<object>* + is/are *<relationship>* *<location>* + *<temporal>* then *<action>*+

Where *<relationship>* is “entering,” “leaving,” “in,” “alone in,” etc., and where *<location>* is one or more offices, conference rooms, open areas or hallways (the selection of multiple locations is interpreted as a disjunction), or exactly one of the following special cases: “AnyWhere” or “NoWhere.” See **Appendix C** for the “Event Specification Language (and Interface).”

2.2.3 Use Case 3: Point/Place of Interest (POI) (Mobile Objects)

“I want to see a map of my location and who is around me…”

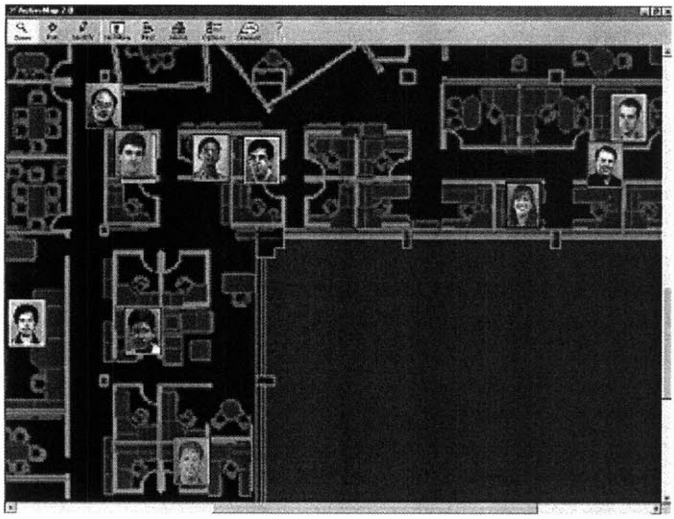


Figure 2.4: ActiveMap (Image Source: AriaView)

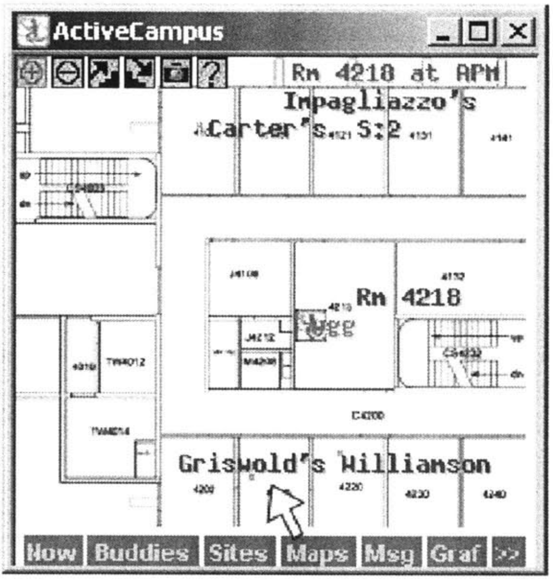


Figure 2.5: The Map and Buddies services (Image source: Active Campus)

Infrastructure Types Characteristics Overview	
Scale:	Floor/room
Infrastructure type:	
1. Communication	
Network type	Service-based
2. Positioning	
Positioning type	Absolute
Accuracy / Precision	Room-level / Few feet
3. Mapping	
Data model type:	Geometric
Modeling language	Needs a location component
4. Software (Services)	
Architecture-type	Closer-architecture
Service type	"Location-aware"

Table 2.3: Use Case 3 – Infrastructure Types Characteristics Overview

Infrastructure Requirements and Potential Use of Standards

Similarly as the previous use cases, a seamless handover for both communication and positioning (location) is a requirement. The significance of this use case is not only tracking mobile users in real-time, but displaying their location on a map in real-time. This requires the software platform (**Chapter 6**) to be scalable to more users using the service.

2.2.4 Use Case 4: Mobile Commerce ("Product Finder," "Personalized Shopping Assistant")

> *(1) "I want to find a specific product (and its store) that I know of...(this product might or might not be on sale)"and (2) "I want to be notified when I am in the vicinity of that store, which has the product I am looking for..." or (3) "I want to be notified about the product that I am looking for *only* when it is on sale. I might allow the service to use my User Profile to target me for similar products, special promotions, etc."*

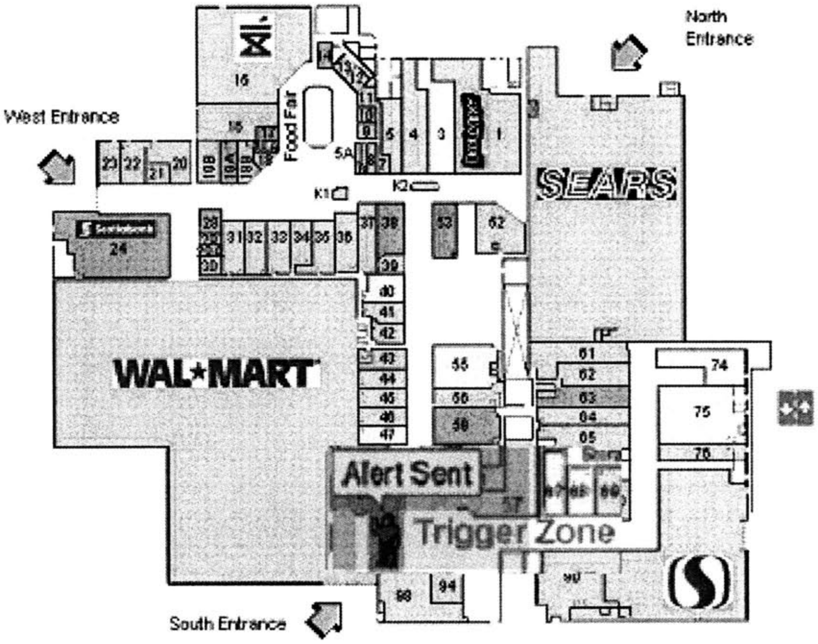


Figure 2.6: 'Alert-Based' or 'Personal Trigger' LBS Service

(1) The user is interested in locating a particular store that has a product of interest. The user can ask either for the "*nearest*" (proximity) store or "*pinpoint*" a store. The former case is when the user wants to know "what is the location of the nearest store, of a known type?" Following this, a proximity search for the nearest store is executed against a Yellow Pages Server. The latter case is where the user identifies the store by specifying its name or the type of business that it is (the products that it sells). If known, the user could also specify the store by a phone number or some other unique ID. Following this, a search for the specific store location is executed against a Yellow Pages Server. For both cases, then, the fetched location of the store is displayed on a map on the mobile device, relative to the known device location. Optionally, the user can then request the best route to the store, display directions to the store (or dial the store).

(2) The alert service is used by a store that has the product to notify interested customers who gave access permission rights to the service provider (i.e., stores in the mall or a third party). Users will receive a message about the sale as they enter the vicinity of the store. Therefore, customers will receive the message before they start shopping and not afterwards, giving the retailer a competitive edge.

(3) This is a special case of the previous action. Here, the user wants to be notified only when the product is on sale. This could be a 'live' service where the user bargains/bids (similar to Ebay.com or Priceline.com) with the store vendors, letting them know what he/she is interested in buying and the price via a user profile. This is a dynamic service not only because the user's position changes as he/she moves around the mall and being target by vendors based on their locations, but also due to the prices of products changing "live" or on-the-fly. Only the stores that meet this price requirement would be allowed to notify the user.

A real-world example of such an indoor LBS application is the "Personalized Shopping Assistant[22]" ('commerce' application category). The current methods of advertising in stores rely on public address systems or programmable LED displays in the aisles. These methods are not effective due to the noise, distractions and the impersonal nature of the method. Shopping by catalog or from home (such as using Prodigy or TV-based home shopping networks) has great appeal but is not going to replace physical shopping. Also, mobile text-messaging (or SMS) allows companies to distribute advertising messages to mobile phones, targeted by user preference profiles. What customers are frustrated about is where to find things, determining the right price, etc.

Overall, location will soon be an aspect of this targeting, limiting the campaign to people currently in a specific location. LBS services are more refined and, in turn, will have a higher value on return. Coupled with this can be the concept of an e-voucher within the message, to enable further enticements of discounts in the store for other products. Advertising messages are another marketing channel to customers and therefore have the potential of very high returns for the communication infrastructure provider (i.e., wireless carrier / operator). Therefore, these providers will eagerly offer this service widely to its clients.

The Personalized Shopping Assistant is based on two products: a very high volume hand-held wireless communications device, the PSA (Personal Shopping Assistant), that the customer owns (or may be provided to a customer by the retailer), and a centralized server located in the shopping center to which the customer communicates using the PSA. The centralized server maintains the customer database, the store database and provides audio/visual responses to inquiries from tens to hundreds of customers in real-time over a small area wireless network. By integrating wireless, video, speech and real-time data access technologies with location positioning, a unique shopping assistant service can be created that personalizes the attention provided to a customer based on individual needs, without limiting his movement, or causing distractions for others in the shopping center.

The objective of this service is to enhance the shopping experience of users by exploiting their contexts in a store. Each shopper carries a specialized device. As the shopper wonders around in the store, the device automatically displays the description of the items that the shopper is currently seeing (location context). In addition to helping the user locate an item, the service can also recommend sales items that match users' interests without any explicit user instructions (personal context) and do a comparative price analysis. As a matter of fact, the MSN Shopping

---

[22] Asthana, A., Cravatts, M., and Krzyzanowski, P. An indoor wireless system for personalized shopping assistance. In Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pages 69-74, Santa Cruz, California, December 1994. IEEE Computer Society Press.

Assistance[23], while not location-based, offers a live chat with a real-person shopping assistant that can help the user find a great gift for anyone on the customer's list. Also, Microsoft's OnSale Mall Buddy service has personalized sales announcements, i.e., "*Alert me when electronics are on sale*"). These applications can make use of yellow pages services to request content of interest (COI).

Another use for being able to locate and track customers is for commerce, analyzing customer shopping behavior. Shopping malls and individual stores get very valuable information about exactly where each visitor goes, which can help with improving store layout, marketing, etc. They could also communicate with customers using screens located around a store – when someone walks in front of a screen it could display information known to be of interest to that person (by means of accessing that person's User Profile saved on his/her mobile device). This would enable stores to combine the advantages of a personalized recommendation system like that used by Amazon.com with the advantages of a physical store, such as the ability to physically inspect merchandise and take it with you immediately. With respect to profit making, m-coupons, cited as a strong growth area in industry reports, will remain niche markets with slower than expected growth[24].

Infrastructure Types Characteristics Overview		
Scale:		Building/floor
Infrastructure type:		
1. Communication		
	Network type	Service-based
2. Positioning		
	Positioning type	Absolute
	Accuracy / Precision	Aisle-level / few feet
	Orientation technology	IR directional beckons
3. Mapping		
	Data model type	Geometric
	Modeling language	Needs a location + event components
4. Software (Services)		
	Architecture-type	Closed-architecture
	Service type	"Location-aware" (proactive)

Table 2.4: Use Case 4 – Infrastructure Types Characteristics Overview

Infrastructure Requirements and Potential Use of Standards

This is use case demonstrates the need for high accuracy levels that will allow to locate a user, for example, in an aisle of a store. The option is to deployment such an LPS system like MIT Cricket (**Chapter 4**) or have the customer attach a sensor tag like UbiSense's UbiTags (**Chapter 4**). However, attaching these tags to their personal mobile device might be a problem (due to privacy issues, as well as technological, i.e., software, issues). The alternative would be a custom

---

[23] MSN Shopping Assistant: http://shopping.msn.com/softcontent/softcontent.aspx?scpId=3565&scmId=1422

[24] 3G: http://www.3g.co.uk/PR/Oct2003/5950.htm

store tagged-device attached to each shopping-cart. This concept was developed by the "Personalized Shopping Assistant," and is portrayed in **Figure 2.7**.

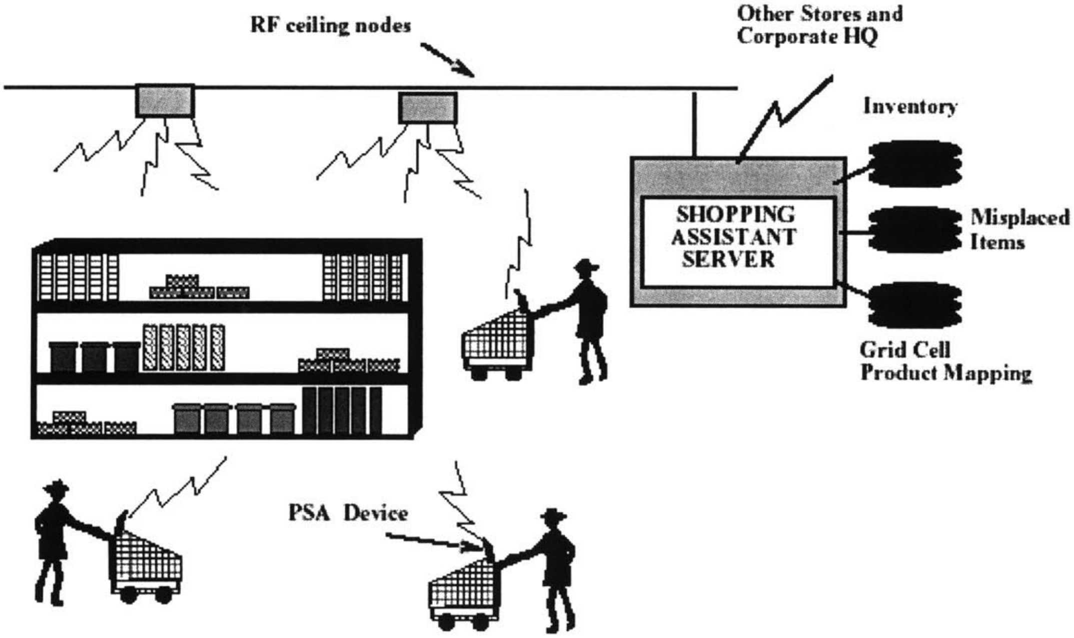


Figure 2.7: Personalized Shopping Assistant (Image Source: Asthana, A, et. al.[25])

In terms of the software (service) infrastructure, these type of trigger services (proactive) are required to push response targeted at the user when his/her location changes. For this to work, an event service (server) is needed to enable this dynamic activity generation. The concept of tailored special offers from stores nearby also applies here, where stores could compete for customers with last-minute deal offers. An Event Server and OnSale Server (**Chapter 6**) would be needed to enable this interaction "live."

Also, a virtual infrastructure can exist where a mobile user would use 'Phonemarks' to store a content of interest (i.e., a product) for later use (i.e., mobile commerce) on the Web. Detailed geo-referencing and the application of 'phonemarks' opens the possibility for querying objects such as products in a shopping window or a painting in a museum. YDreams' FluidShopping was designed to reduce the anxieties of shoppers through the use of Internet enabled mobile phones, helping find the product they want to buy and enabling its purchase after hours from the shopping window.

2.2.5 Use Case 5: Navigation

(1) "I want to be navigated to the store that has a product on sale…"and (2) "I want to be notified along the way when I am near someone (or something) …"

---

[25] Asthana, A., Cravatts, M., and Krzyzanowski, P. An indoor wireless system for personalized shopping assistance. In Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pages 69-74, Santa Cruz, California, December 1994. IEEE Computer Society Press.

(1) Having two (or more) positions (the user's initial location and the location of the destination) the service can calculate the route between them. The service would compute the route (best/shortest) between the points or "spaces" and display the route (map) and/or directions (text). Typically, the route will be determined between two locations, one being the starting point or "space," which is the current device location, and the other being the end point (the destination).

(2) There is also the case where the user has several stops along a route (i.e., a store that might have something on sale that is of interest to the user or a friend that might be in the vicinity) so the user must specify waypoints in addition to an endpoint.

The endpoint and waypoints may be determined by any number of ways. For example, the user might use the *'pinpoint'* or *'nearest'* (proximity) services described above to establish these points or spaces ("nodes"). Once the route has been calculated, the user has two options:

- Display route: show a map of the floor plan layout.
- Display directions: this might as important as maps and actually preferred by some users as it may be easier for them to follow instructions than to read a map.

A real world application is the WayFinder from MIT Cricket. This application, running on a handheld computer, can help sighted or blind people navigate toward a destination in an unfamiliar setting. For example, the WayFinder might lead a person from the entry lobby of a building to the office of the person hosting the visitor or to a seminar room. The WayFinder gives incremental directions to the user on dynamically active maps using the user's position and orientation with respect to a fixed set of wireless beacons placed throughout the building.

In addition, when considering a navigational application, a wheelchair or the Segway[26] can be considered as the user's mobile device, which can be designed to (automatically) navigate the user to his/her destination.

Infrastructure Types Characteristics Overview		
Scale:		Building/Floor
Infrastructure type:		
1. Communication		
	Network type	Service-based or ad-hoc based
2. Positioning		
	Positioning type	Absolute or relative
	Accuracy	Room-level / Room-level
3. Mapping		
	Data model type	Symbolic or geometric
	Modeling language	Needs a location + navigation + event
4. Software (Services)		
	Architecture-type	Closed-architecture
	Service type	"Location-aware" (proactive)

Table 2.5: Use Case 5 – Infrastructure Types Characteristics Overview

---

[26] Segway (the human transporter) http://www.segway.com/

Infrastructure Requirements and Potential Use of Standards

In terms of the positioning infrastructure, positioning aids such as spatial code bar (SBC) or emitters are needed around a building. **Figure 2.8** shows an example of such SBSs placed in the Columbo Shopping Center in Lisbon, Portugal. Using Wi-Fi or Bluetooth access points (emitters) as anchors will allow users to automatically pinpoint their location without having to read or write any SBC.

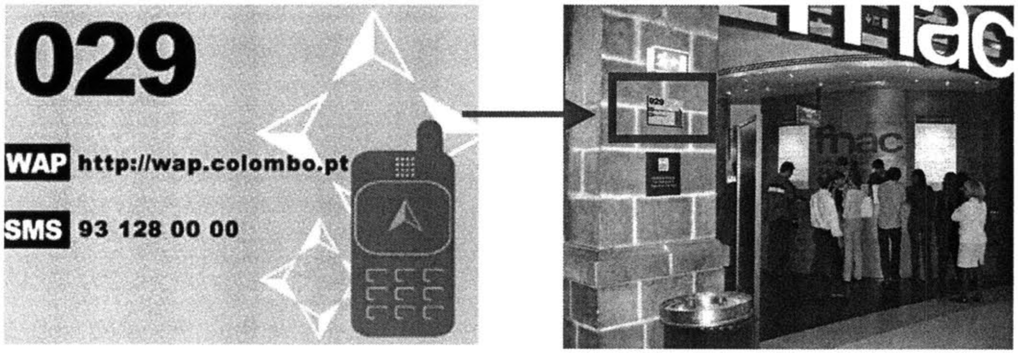


Figure 2.8: Spatial Code Bars. Tag placed on store in Colombo Shopping Center, Lisbon, Portugal [Source: YDreams, Vespucci LBS Summer School]

Moreover, locating a user based on Wi-Fi cell-id might be adequate for navigating the user to a store, but might not be precise enough for a shopping experience inside the store. More precise positioning technologies could enable navigation to specific items within a store's aisle. **Chapter 5** explores navigation in terms of relative versus absolute positioning. Orientation might also be desirable, which can use the MIT Cricket ViewFinder application (see next use case).

The mapping infrastructure is used for exploration and navigation of a place, which is usually associated with the use of symbolic representations such as maps (of a shopping mall) that are then matched against the reality they try to symbolize. This matching is facilitated by the use of 'anchors' (points of interest) and 'paths' displayed in maps that are easily recognizable locations. These maps and its features (i.e., anchors, paths) can be modeled either using symbolic representations or geometric features.

If the service is using a geometric map, the user may optionally display distances along segments of the route. If the service is based on a symbolic model (spatial relationships are either based on associations, containment, or proximity), the directions will not be given precisely as to how many feet before making a right turn. In either case, geometric or symbolic, the directions should probably be given by means of symbolic reasoning (i.e., "when you reach store A, make a right turn," instead of "after 10 feet, make a right turn") as people associate with this type of reasoning better. In the case of the Segway or an automatic wheelchair, directions need to be given in terms of geometric reasoning since the machine is not capable of symbolic reasoning. **Chapter 5** explores geometric versus symbolic location models in terms of positioning and navigation.

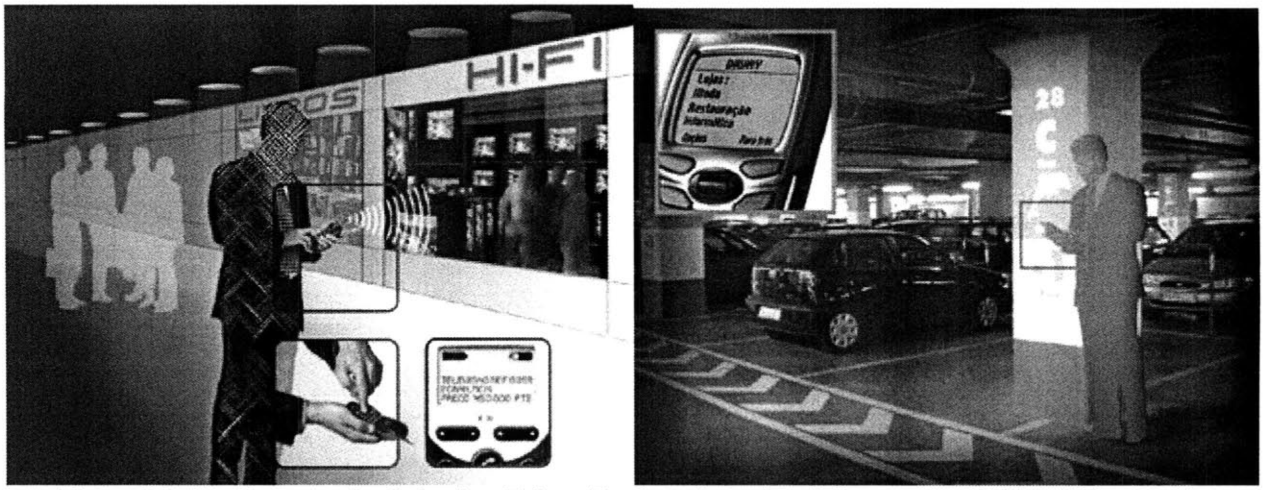


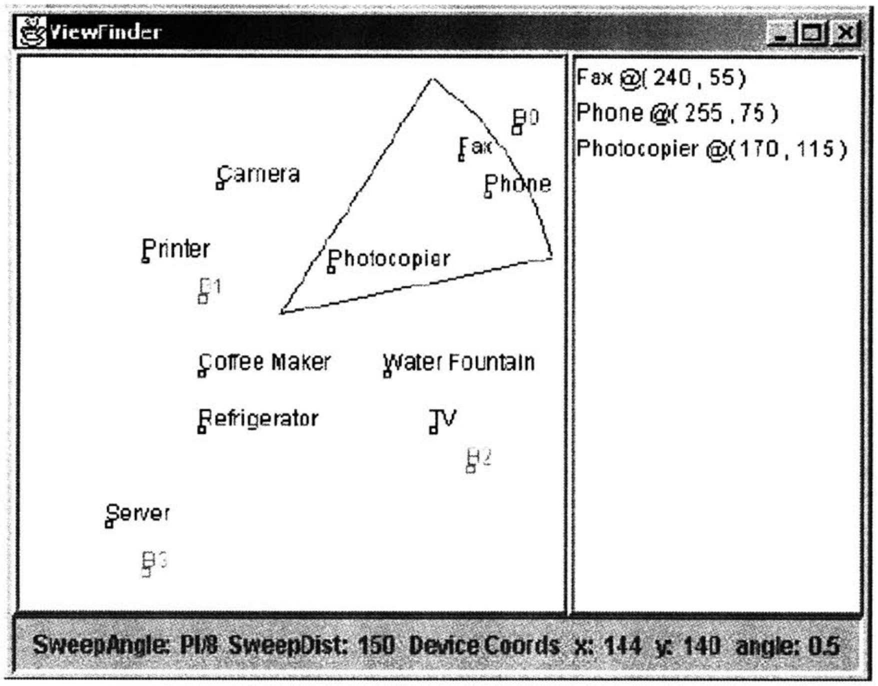
Figure 2.9 a/b: Phonemarks & Anchor Points: Product Querying & Interaction (left) and Storing the Location of a Car (right)
[Source: YDreams, Vespucci LBS Summer School]

Overall, indoor navigation is going to be more often the desire than the need until an infrastructure of positioning/navigating aids (i.e., anchors) is in place.

In terms of the software (service) infrastructure, the user will most likely invoke a location-based concierge application that would determine his/her location and he/she would subsequently receive navigation/routing details about getting to the destination Optionally, the service is able to specify some waypoints (or these points have be specified by the user) and the user may specify route determination criteria. The criteria might be: fastest, shortest, etc., and can also specify the preferred mode of transport of the user (elevator vs. staircase). The details of the returned information might include directions to the site and other relevant information (according to the user's profile, for instance). The route can also be optionally stored on the terminal or application server. The user may store it for as long as needed, thus requiring the means to also fetch a stored route. Regardless of how the endpoints and waypoints are established, the information is then sent to a Route Server (**Chapter 5**) that calculates the route. Applying OGC's OpenLS services, a subscriber would seek navigation advice from the service provider via a Personal Navigation Service. This is an application service that utilizes OpenLS Core Services (Gateway, Directory, Geocoder, Presentation, and Route Determination), discussed in **Chapter 6**.

2.2.6 Use Case 6: Viewfinder

"What services/devices are available within my sweep angle?"



```
Fax @( 240 , 55 )
Phone @( 255 , 75 )
Photocopier @( 170 , 115 )
```

```
SweepAngle: PI/8 SweepDist: 150 DeviceCoords x: 144 y: 140 angle: 0.5
```

Figure 2.10: ViewFinder Application (source: MIT Cricket)

Users should be able to identify features (i.e., products in a store) of their surrounding by just pointing at them. For this reason, special sensors have to be integrated into the user's mobile device that allows the determination of directions. With MIT Cricket ViewFinder application, the user can point in any direction and specify a "sweep angle" and maximum distance. Using an "active map" integrated with a resource discovery system (i.e., MIT's INS), the ViewFinder then retrieves and displays a representation of the set of devices and services located inside the sector of interest specified by the user. It then allows the user to interact with these services via the representation on the map.

To enable this functionality, the ViewFinder application using the compass software queries the INS resource discovery server to obtain the global coordinates of the available services. To facilitate the bootstrapping process, the name of the server for the space is advertised on the RF channel by the beckons. The Cricket system also assumes that individual services use their own software compass to obtain their coordinate information, and that they advertise this information to the resource discovery system. Otherwise, a system administrator can assign global coordinates to each individual (static) service.

Infrastructure Types Characteristics Overview	
Scale:	Room
Infrastructure type:	
1. Communication	
Network type	Service-based
2. Positioning	
Positioning type	Absolute
Accuracy / Precision	1-3 inches
Orientation technology	MIT Cricket's compass software
3. Mapping	
Data model type	Geometric
Modeling language	Need for 3D
4. Software (Services)	
Architecture-type	
Service type	"Location-aware"

Table 2.5: Use Case 5 – Infrastructure Types Characteristics Overview

Infrastructure Requirements and Potential Use of Standards

The one striking requirement that is different from the other uses cases is the need for 3D to model the view shed angle. 3D is also of use to a user who wants to pinpoint his/her location or see where the "buddies" or products are located in relation to his/her position.

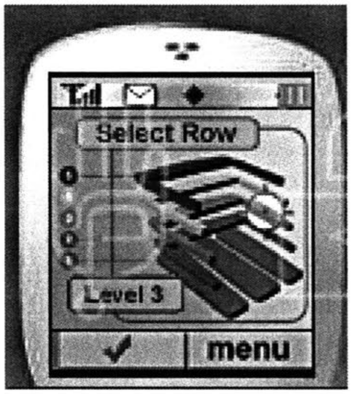


Figure 2.11: 3D Display of the Indoor World

2.3 Conclusions

The advances in indoor location positioning technology (**Chapter 4**) which are now being introduced to the market are enabling a number of indoor LBS applications like the ones featured in this chapter. The core functionality of these applications is similar across most LBS applications: find an object, person, or place/event. The most common or popular applications fall into the fallowing categories, presented in **Table 2.6**.

Use Case Summary		
Use case	Name	Description
1	Point/Place (Area) of Interest (POI)	"where am I?" or "what's (static objects) / who's (mobile objects)around me?"
2	Location-based triggers	"Buddy Finder," "Conference Assistant," "Child-Alert"
3	Mobile commerce	"Product Finder," "Shopping Assistant"
4	Navigation	text, voice, or map directions
5	Viewfinder	3D orientation

Table 2.6 Summary of Use Cases

While a good start, these standalone ("stovepipe[27]") applications are unlikely to make a large impact on the marketplace, for a number of reasons. First, these applications are currently implemented in a vertical way with 'full-systems' and 'closed-architecture' characteristics. Moreover, there is no consensus on common standards for designing these applications, and, as a result, there is no common software platform on which to build such applications that would enable easier deployment as discussed in **Chapter 6**.

Moreover, the heterogeneity stemming from this situation results in a lack of interoperability, which prevents application integrations and service-chaining (**Chapter 6**). This means that there is no way for an application to share, access, or control the sensing resources without knowing the sensor and the network specifications. This situation will predominate for some time until the understanding and language required to define open and standardized interfaces at each level (pertaining to each infrastructure type) of the system has been developed. This heterogeneity stresses the significance of the standardization process and the role of open interoperability standards, which is addressed next, in **Chapter 3**.

Then, each of the chapters on the infrastructure types explores in detail how each level of the system can be abstracted/standardized. In **Chapter 4**, we address the LIF MLP API, which is a standardized application-level protocol for querying the position of mobile users independent of the underlying communication infrastructure (i.e., wireless network). In **Chapter 5**, we address how this position data can be modeled in a standard way using GML, as part of the mapping infrastructure, which is about two things: (1) modeling the indoor world in terms of its geographic and symbolic features; and (2) modeling or overlaying the user position data within this overall model or representation about the indoor world. In **Chapter 6**, we address the software (services) infrastructure, outlining the various software components and their interaction (interfaces) to enable "simple or basic" or "location-aware" LBS services. More importantly, we address how the role of open interoperability standards for interfaces and architectures is of great importance with respect to enabling these applications to share, access, or control the sensing resources being independent of the underlying communication infrastructure, as well as the positioning infrastructure.

---

[27] "Stovepipe" is a metaphor commonly used to describe systems that are integrated "from top to bottom" but isolated laterally, i.e., from other systems. A stovepipe system might be a system from a single vendor or it might be a system built by an integrator, but it is not an open system.

In **Chapter 7**, as part of the business scenarios, we clarify the misassumption about niche market applications, which is that there appears to be no single ““killer” application The natural tendency is to forget that several niches added together can constitute a very significant market in total. Still, the successful business model for indoor LBS will depend upon the specific niche (target) market and applications involved, not a one-size-fits-all approach. Overall, this goes back to the dilemma problem of needing both the infrastructure and the “killer” (or at least a valuable) application. Without the application people will not invest in the infrastructure, and without the infrastructure the open-market for iterating towards valuable applications and their business models will not exist.

In **Chapter 8**, we use our framework approach to analyze how these applications can be rolled-out as services considering the many factors and uncertainties involved in the LBS value-chain.

CHAPTER 3

INDOOR LBS: THE SIGNIFICANCE OF THE STANDARDIZATION PROCESS AND OPEN INTEROPERABILITY STANDARDS

3.1 Introduction

Before discussing each of the infrastructure types and their role in deploying indoor LBS applications such as the ones we reviewed in **Chapter 2**, we decided to first address the significance of open interoperability standards, which in effect, function as the glue between the four infrastructure types and the components within them. Without this glue, implementing such applications is complex and presents challenges (i.e., barrier to market entry) impeding the growth of the industry to its critical mass level.

Currently, most of the indoor LBS applications are based on self-contained systems (vertical implementation of all the necessary components such as location positioning servers, middleware, applications; all are in one package from one vendor) that provide a one-size-fits-all solution. The optimum strategy must be to use an open architecture and platform that is capable of uniting and integrating different features and functions from various infrastructure providers in a highly distributed way to create a rich variety of different value propositions for a diverse customer audience. We elaborate on this strategy more in **Chapter 6**.

The purposes of this chapter is to introduce on how well the standardization approaches of LIF/OMA and OGC can serve the use cases and scenario development discussed in **Chapter 2** and **Chapter 6**, respectively. Through out this thesis, the issue of interoperability needs are addressed across the four infrastructure types (the next three chapters), in addition to business strategies (**Chapter 7**) such as incremental growth.

3.2 Interoperability and the Role of Standardization

> *"Standardization is one of the essential building blocks of the Information Society... The development and use of open, interoperable, non-discriminatory and demand-driven standards that take into account needs of users and consumers is a basic element for the development and greater diffusion of ICTs and more affordable access to them..."*
> *(World Summit on Information Society (WSIS))*

The process of standardization has been important in creating and growing global markets for computing and communications systems. For example, communication network standards like GSM, Ethernet, and IEEE802.11 enable interoperability between equipment from different manufacturers, lower costs, and reassure users that their investment in technology will be viable beyond the short term. However, since the market for indoor LBS is in its infancy and the requirements for indoor LBS application are just beginning to be understood, caution must be taken to avoid making early decisions that will impede market adoption. In **Chapter 8**, we analyze certain market conditions and the factors (filters) that affect such decision-making.

Standardization can be a serious success factor for any new technology, and LBS are not an exception to this rule. Standardization activities for LBS technologies should be rooted in the market because of the huge value to the overall market in the long term. When standards are adopted, the ultimate technical benefits will result through interoperability between systems and software from different vendors, allowing for the reuse and exchange of quality data products, with seamless integration of the location information, into any existing network infrastructure. From the business point of view, standards lower costs and reassure users that their investment in technology will be viable beyond the short term.

For example, the process of standardization has been important in creating and growing global markets for computing and communications systems. Standards like GSM, Ethernet and IEEE 802.11 enable communication network interoperability between equipment from different manufacturers, lower costs, and reassure users that their investment in technology will be viable beyond the short term.

In terms of the software (services) infrastructure, interfaces and protocols should be published (open) and standardized as a general business and technical requirement. This would reduce the engineering complexity of communication among LBS content/service providers as well as developers. Open interoperability standards support the commercial viability of a LBS service provided by a community of cooperating yet competing LBS providers. Without interoperability standards, application domains will continue to remain a standalone implementation (niche markets), when there is the potential for a broader market with the bundling of services (**Chapter 6**).

A standardized API for location data access from any positioning technology (**Chapter 4**), for example, would enable interoperability among distributed indoor LBS applications and services. In the context of geosptial information and services, there are standards dealing with the manipulation of geospatial data (spatial standards, providing interfaces for standard methods), but also standards for access to geospatial data – metadata standards – that are used for catalogues (or search engines) that describe the content of information resources; they are needed to be able to search and retrieve data and service (i.e., OpenGIS Web Service[28]) resources in the Web and, as a result, they need to be in standard formats. As a result, collaboration or service chaining among the different application is possible (**Chapter 6**).

Nevertheless, considering the fact that standardization can also be a failure factor, care must be taken to avoid making early decisions that will impede market adoption. For example, the Magic Services API is the fourth method to get position information from a network. Unfortunately, in contrast to the other methods, Parlay, LIF (which we apply in **Chapter 4** for the positioning infrastructure), and WAP APIs, the Magic API had no providers of mobile positioning centers committed to implementing it. Magic, which stands for Mobile Automotive Geo-Information

---

[28] OpenGIS Web Services (OWS): www.opengis.org/ows

Services Core, was created by a loose industry group, led by Microsoft, and biased toward the automotive industry.

Another example is the failure of WAP in Europe, which is mainly because the technology was over-sold (due to bad marketing) to the detriment of services. Another cause comes from relationships between operators and service providers/content providers: mobiles operators did not yet enter in agreement for resource sharing (see scenario 3 in **Chapter 7**). In Japan, the success of mobile services is mainly due to the cooperative business model between NTT DoCoMo and its service providers for I-mode. The Japanese network operator deducts a commission of 9% on services offered on its portal.

Even though the indoor LBS market is in its infancy – application requirements are still not well understood – the indoor LBS field must settle on a few, consensus-derived and well proven standards and practices. This is especially true considering the general fact that LBS services are dependent not only on a number of direct enabling technologies, but also on a number of indirect facilitating technologies of added value services. The need for a focused effort concerning location interoperability is also evident considering the multi-layered nature of the LBS industry responsible for developing, operating, and supporting the location services value chain. Hence, interoperability among indoor LBS systems and applications should be attained as a result of a coordinated effort of these diverse players. For this reason, OGC's OpenLS initiative (discussed below) brings together key industry players to build and consolidate the standards infrastructure for these interoperating LBS services.

Moreover, while standards are important, it should also be recognized that these are often more accurately "discovered" than imposed, evolved and adopted as a result of real world pressures, rather than through a top-down process. Standardization processes are vulnerable to pressure from large vendors with their own particular interests to support, and for communication infrastructure providers these factors can in turn limit their own freedom to innovate and respond rapidly to threats and opportunities.

3.3 Standards Framework

There are many standards organizations (see **Figure 3.1**) such as the Internet Engineering Task Force (IETF)[29] and The World Wide Web Consortium (WC3)[30] that have location and geo-spatial initiatives, working groups, and published specifications, yet location is not their core focus. Even the proposed ZigBee (IEEE 802.15.4) standard[31] for ubiquitous computing represents one current effort towards the goal of location awareness that is already being discussed in the context of that proposed standard. In addition, many of these large organizations have slowed adoption of location-specific standards as a result of organizational member's unwillingness to support standards, which may not support their business strategies.

---

[29] IETF: http://www.ietf.org/

[30] W3C: http://www.w3c.org/

[31] ZigBee Alliance website http://www.zigbee.org/

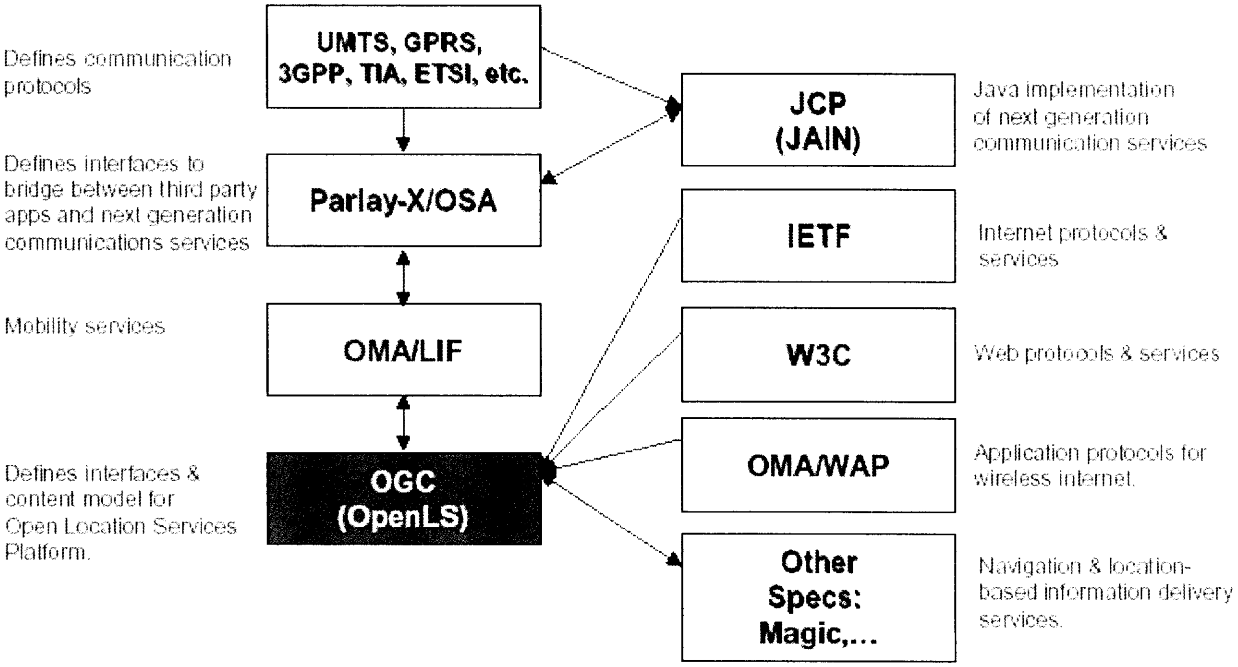


Figure 3.1 Standards Organizations with Location & Spatial Initiatives (Image Source: OGC)

To offset this lack of focus on location on both the wireless and Internet fronts, two well-positioned organizations have emerged as the drivers of location interoperability: the LIF[32] and the OGC[33]. Both OGC and LIF are seen as the standard holder regarding wireless location interoperability that ensure the smooth flow of information between content repositories, the Internet, and end-user devices through open protocols. These two organizations are accomplishing this task through a cooperative strategy of working with other standards bodies (shown in **Figure 3.1**) to promote a cohesive set of wireless location interoperability standards.

As a result, standards will provide application and content providers with a ubiquitous contextual meaning to location, regardless of the location positioning technologies utilized. As standards bodies are continuing to define network nodes, functionality, and interfaces, communication infrastructure providers (i.e., mobile operators) face the reality of integrating their location servers with network elements that have varying levels of standards compliance and back-end systems with unique interfaces. In short, no "one-size-fits-all" solution will work.

3.3.1 OpenGIS Consortium (OGC)

OGC's focus is on the application, data and presentation layers of the Internet stack. The types of services that are fundamental to any spatial data infrastructure (SDI): data catalogues, online/mobile mapping, and access. Other services include coordinate transformation,

---

[32] LIF's 140+ members include Ericsson, Motorola, Nokia, Siemens, SignalSoft, and Autodesk. LIF website: www.locationforum.org

[33] OGC's 250+ members include Hutchison 3G UK, Oracle, Sun Microsystems, ESRI, and Vodafone. OGC website: www.opengis.org

classification, data authentication and validation, data analysis, data fusion, custom symbolization, multi-person collaboration, gazetteers, processing algorithms, and service catalogues allowing discovery of required services (see more in **Chapter 6**).

In the LBS market, OGC accomplishes this through the OpenLS initiative[34], which engineers location services application interfaces designed to support interoperable solutions that "geo-enable" the Internet and wireless location-based services, and mainstream IT. The OpenLS initiative focuses on developing interface standards "needed by industry to support implementation of the location services invoked by mobile or wireless Internet devices in end to end settings" (Hecht, 2000)[35]."

The OpenGIS Implementation Specifications focus on the functionality a software component (such as a map viewer) should support, and the interfaces required to connect to, and extract data from, such a component. The specifications do not specify implementation details, meaning, that a component should be coded in C++ or Java, be specific to an operating system, etc. Similar to the way that HTTP protocols enabled the growth of activity in the World Wide Web, OpenLS standards, resulting from OGC's cooperative testbed process, have the potential to enable significant growth of Location Services markets in the Wireless Web" (Burnett, 2000)[36].

3.3.2 Location Interoperability Forum (LIF)

In contrast to OGC that deals with the application side (geo-processing), LIF, now part of the Open Mobile Alliance[37], focuses on interoperability from the wireless side of the LBS market. LIF is dealing with specifications pertaining to the query and response for the actual location or position of the mobile device. The vision of LIF is that LBS service are seamlessly integrated and available to all mobile users wherever they are.

LIF has developed the Mobile Location Protocol (MLP) standard API for utilization by carriers, wireless infrastructure providers, and mobile application developers. The role of MLP is to seamlessly integrate location from the location carriers/operator communication network (see more on how the LIF MLP API is used for the communication/positioning infrastructure in **Chapter 4**). Moreover, MLP is an application-level protocol for the positioning of mobile terminals and is independent of the underlying network technology, and, in turn, the position method.

---

[34] OGC's OpenLS: www.openls.org

[35] Hecht L. (2001) Open Location Services: Vision and Objectives, Corporate Roles and Work Program Summary, http://www.openls.org/docs/OLS_Oveview.htm

[36] Burnett J.(2000) OGC Announces Wireless Location Services Initiative, http://www.openls.org/News/00-10-30.htm

[37] Open Mobile Alliance: http://www.openmobilealliance.org/

From LIF MLP specification abstract[38]:

> *The purpose of this specification is to define a simple and secure access method that allows Internet applications to query location information from a wireless network, irrespective of its underlying air interface technologies and positioning methods.*
>
> *This specification covers the core of a Mobile Location Protocol that can be used by a location-based application to request MS location information from a location server (GMLC/MPC or other entity in the wireless network).*
>
> *...The API is based on existing and well-known Internet technologies as HTTP, SSL/TLS and XML, in order to facilitate the development of location-based applications.*

LIF was set up in 2000 to make sure that the location industry did not fragment into a number of incomplete technology islands, specifically to address the growing concern of location-centric barriers to wireless interoperability. This was especially urgent when considering the question of how application servers could address different mobile position gateways (or work directly with the HLRs). Otherwise, an application server might need a different way of connecting to particular manufacturers' network, despite having all of them use the GSM network. LIF has agreements with other organizations such as 3rd Generation Partnership Project (3GPP)[39] and WAP Forum (WAPF)[40].

Interoperability between LIF and OGC standards is critical when it comes to the convergence of location technologies and the widespread adoption of end-to-end solutions for LBS. LIF and OGC work closely to ensure that the Internet and wireless standards driven by either organization are mutually supported. Specifically, OGC works closely within the LIF organization to ensure that the LIF Document Type Definition (DTD) and OpenLS XML schemes work in harmony, allowing content transmission between LIF and Open LS based software without the loss of information and minimal or no translation. This is critical to supporting the wireless to Internet interchange of location-specific content and service support.

Even though there is considerable overlap between LIF and OGC concerning wireless location interoperability, it provides for expanded awareness and adoption of the combined standards.

3.4 The Success of Standards – Adoption

Standards adoption is a means to an important end – building critical mass in the development of interoperable data and services. Note that critical mass here means that the creation of many diverse interoperability solutions will not necessarily improve the situation with standalone ("stovepipe") application integration.

---

[38] LIF MLP specification: http://www.openmobilealliance.org/tech/affiliates/lif/lifindex.html

[39] 3GPP: www.3gpp.org

[40] WAP Forum: www.wapforum.org

One of the most immediate challenges to adoption of standards is the market inhabitation (or the potential of it) due to individual company's proprietary interfaces (see Scenario 2 in **Chapter 7** where we explore this possibility). The question of concern is at what level is standardization required, and at what level is the technology implementation left up to the vendors. There needs to be room for those "killer applications." Combined with the disparate wireless, LPS systems, Internet and GIS systems, the ubiquitous wireless location interoperability standard will be a long way off from industry-wide adoption. In any case, without standards, there can be no end-to-end location implementation utilizing the best-in-class technologies.

To overcome this challenge of market inhabitation that has the potential to impede the adoption of standards, let us not forget about the end users (the customers). The users of these applications are not concerned with the underlying architecture or technologies powering their LBS user experience. Location information by itself has no value to them. Instead, these customers will welcome the use of location technologies in their everyday lives, provided that it is as mobile, "seamless," and ubiquitous as their lifestyles.

Again, the only way for the indoor LBS industry to meet this challenge is if it works together through the development and adoption of location wireless interoperability standards. The intention of this chapter was to introduce the technical details of current interoperability-related standards. However, the overall scope in terms of adoption should be to raise awareness of the need for, and the potential of, implementing these standards as an avenue toward interoperability.

3.5 Conclusions

It is easy to see the cumulative nature of standards--it is impossible to conceive of a TCP/IP standard without a multitude of other standards that make such communication even possible in the first place. What we are primarily concerned with is those fields where a standard is non-existent, immature, less than satisfactory, or where competing standards exist. We are interested in questions like:

- Does the existing set of practices or evolving standards of LIF and OGC meet the developers needs?
- Do the developers have the clout and/or resources to successfully create or contribute to a new standard?
- Would these efforts serve the developers' business ends?

Perhaps the answers to these questions are ultimately personal and need to be found through discussion and exploration at a personal or company level. However, these pursuits will be aided by a thorough understanding of the process, which is what we will discuss in the rest of this thesis. In short, the question of "why have standards?" is not difficult, and can be answered with minimal reflection. We know that in the long run standards are beneficial. In fact, it is arguable that standards are inevitable products.

Obligation or not however, a great deal of financial and human capital must be spent in standard creation. Few companies or groups can afford to do this merely for unselfish reasons; companies need clear, compelling rationale why their contribution to or adoption of a given standard is a wise use of their valuable resources. IT managers need to be able to develop this rationale as part of the business case in a development project. The following discussion should help managers prepare that case, specifically concerning the standards of LIF and OGC. We address these issues in **Chapter 8**, but first, we explore how standards from LIF and OGC apply to each of the infrastructure types.

CHAPTER 4

THE COMMUNICATION INFRASTRUCTURE & THE POSITIONING INFRASTRUCTURE

4.1 Introduction

The proliferation of lightweight, portable computing devices and high-speed wireless local-area networks has enabled users to remain connected while moving about inside buildings. A multitude of LBS applications such as the ones featured in **Chapter 2** would benefit from indoor positioning and navigation. Because indoor settings have the disadvantage of absorbing and diffusing RF of GPS systems and cellular network systems (i.e., GSM and UMTS), their positioning mechanisms (i.e., TOF, OTR, AOA, etc) are not appropriate to provide the location of a user inside buildings. (There are cases, however, that this might be possible with accuracy levels in the magnitude of tenths of meters.) As a result, indoor LPS systems introduce new/adopted positioning methods outlined in this chapter and explained in **Appendix A.2**.

This chapter reviews these indoor LPS systems, specifically with respect to their positioning output, which is either relative or absolute. The Buddy Finder application can function based on relative positioning where a location-based trigger would alert the user when his/her friend is in the proximity. For this purpose, the Active Badge LPS system is sufficient where positioning is based on being sensed in a location (i.e., a room). However, it is essential to be able to tell with certainty which room a person is in, or which side of a partition they are on.

For other indoor LBS applications, accuracy requirements are more precise than one might think. The Product Finder application can function based on either positioning depending whether the application is required to notify the user using a similar location-based trigger as he/she is in the proximity of the store or pinpoint/navigate the user to the shelf inside the store where the product is located. For the latter case, accuracy to within a foot or so is required; specifically a xy coordinates (absolute positioning). LPS systems such as MIT Cricket can be deployed to achieve this accuracy level.

Navigation can also be in terms of 'relative' or 'absolute,' and it depends on the type of application (use case). Usually, however, indoor LBS applications don't require metrics like distance and time due to the small scale of the indoor space. For example, walking inside a shopping mall doesn't require distance measurements, but a more meaningful representation of location, that is, where things are in relation to fixed objects (see **Chapter 5**).

Note that this thesis does not attempt to determine which LPS will eventually come into more common usage. The primary reason for this is that most of the LPS systems reviewed are currently in the early stages of development and it is simply too early to tell which LPS will provide a more effective and efficient data source for sensing location.

Also, this chapter explores the notion of "seamless worlds." Mobile users would like to stay connected while moving indoors considering the fact that a shopping mall may have multiple communication networks (or "worlds") like GSM and Wi-Fi. Or, a Wi-Fi network might be provided by several providers throughout different zones (or "worlds") of the mall. This concept of staying connected while moving is referred to as seamless communication handover. Also, the concept of seamless positioning (location) handover is explored considering the fact that a shopping mall may be furnished with multiple LPS systems that are of different positioning type: 'relative' versus 'absolute.' In this situation, the LBS service should be able to know at all times the location of the user, regardless of the underlying positioning system (i.e., sensor type, positioning method).

The overall goal of this chapter is to understand where potential open interoperability (geospatial) standards might fit in within the communication and positioning infrastructure. This considers the LIF MLP API for retrieving position information from wireless networks in a standard way. The key to remember is that open interoperability standards enable the integration of standalone application (niche markets), resulting in more value-add to the overall market and a more comprehensive service to the user. However, as mentioned in **Chapter 1**, these LPS systems are 'full systems' that will predominate for some time, until the understanding and language required to define appropriate interfaces at each level of the vertical implementation of these LPS systems has been developed.

Before getting into the specifics of each LPS system, the following discusses relative versus absolute positioning and the notion of "seamless worlds."

4.2 Absolute Positioning versus Relative Positioning

It is critical to realize that the LPS systems can determine either the presence (relative positioning) or a specific position (absolute positioning) of mobile users. For discussion purposes, MIT Cricket and Active Bat are used to explain absolute positioning. An example of a relative LPS system is Active Badge. Another LPS system could position the user based on the Wi-Fi cell-id positioning method. There are two possibilities with the Wi-Fi cell-id approach: one can achieve absolute positioning (the Wi-Fi APs have absolute (fixed) coordinates); while the other, relative positioning (the Wi-Fi APs have symbolic locations, i.e., "room 2, floor 201" "sector 1, hallway Z").

4.2.1 Absolute Positioning

Absolute positioning is when objects have a specific xy coordinate or are positioned as a metric offset from a fixed reference. There is a spatial reference unique to a specific position. For example, globally it is expressed in latitude, longitude, and altitude. Additionally, the attribute of direction complements the absolute position. MIT Cricket as well as Active Bat (shown in **Figure 4.1**) provides absolute positions where a user is situated at x, y, z coordinates.

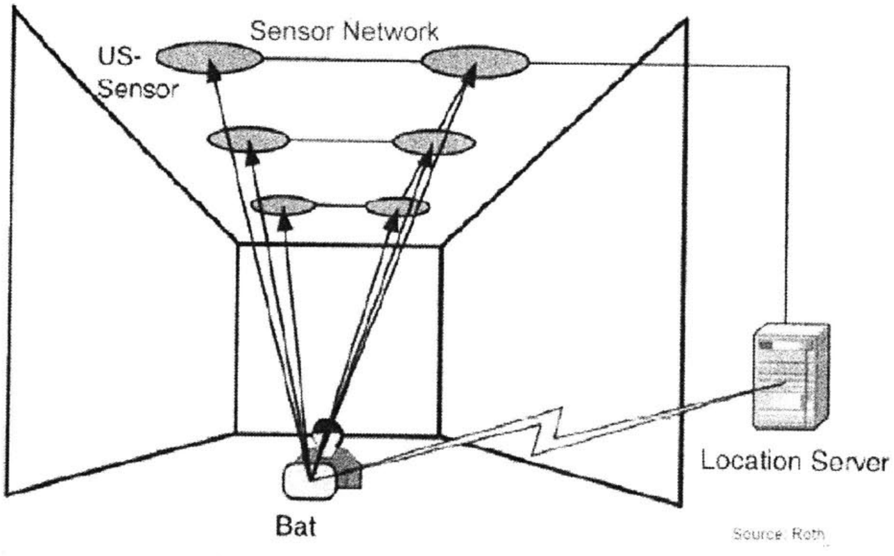


Figure 4.1 Active Bat (Image Source: Roth)

To enable its beacons (sensors) to gather the distance information (ratio of height to distance), MIT Cricket implemented a local coordinate system using four active beacons instrumented with known positions within the space. The beacons are configured with their (x,y,0) coordinates (as shown in **Figure 4.2**) and broadcast this information on the RF channel, which is sensed by the receiver on the compass. Note, that a Euclidean (geometric) model, discussed in **Chapter 4**, makes it possible to define a user's position plus orientation information by a rotation with translation matrix in an orthogonal coordinate referential.

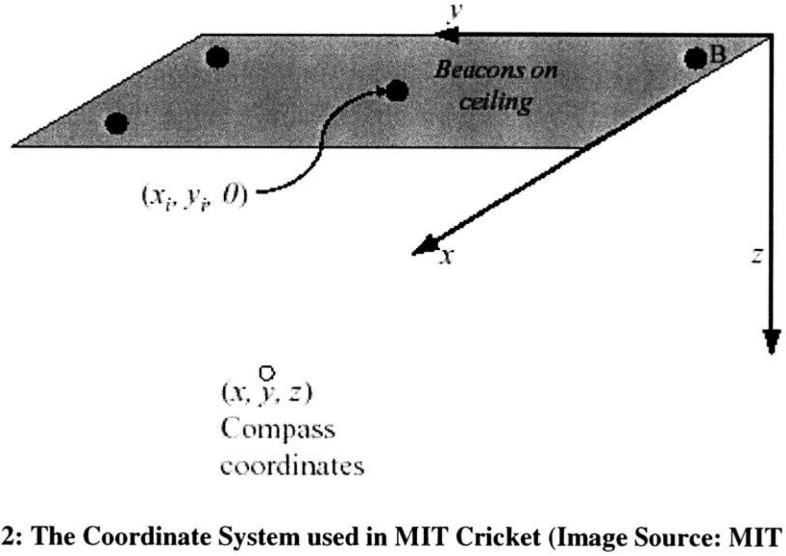
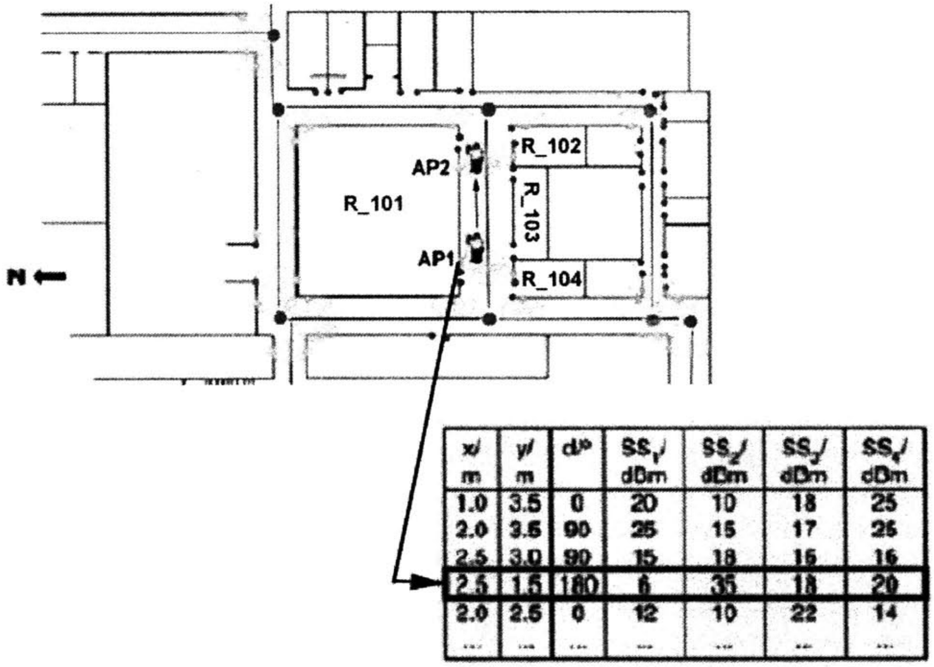


Figure 4.2: The Coordinate System used in MIT Cricket (Image Source: MIT Cricket)

Absolute positioning using the Wi-Fi cell-id positioning method is portrayed in **Figure 4.3** and works as follows. Wi-Fi access points (APs) transmitting RF signal augmented with their physical (fixed) coordinates can be used to estimate the location of the mobile host. The strengths of the RF signals arriving from more APs can be related to the position of the mobile terminal and can be used to derive the distance measurements to position the user. This is an aspect of the cell-id location method used in cellular networks (i.e., GSM). Some rooms may have one or more Wi-Fi APs, but the same referencing approach can be used. The accuracy of the system might be limited by the (possibly large) cell size. Handling off location to a room can be done with a tolerance level ("snapping"). This is applied in the location fingerprinting positioning method, explained later.



x/ m	y/ m	d/°	SS_1/ dBm	SS_2/ dBm	SS_3/ dBm	SS_4/ dBm
1.0	3.5	0	20	10	18	25
2.0	3.5	90	25	15	17	25
2.5	3.0	90	15	18	16	16
2.5	1.5	180	6	35	18	20
2.0	2.5	0	12	10	22	14
...	...	...	...	...	...	...

Figure 4.3 Absolute Positioning

The idea of using "xy coordinates" is that it can give the application developer more flexibility. On the other hand, using relative positions (next section), once the database is defined with relative position symbols such as "*AP1: L_R101; R_R104*" the database cannot be easily modified to change or add additional regions. Moreover, using relative/symbolic position representations limits the logic of an application. If "xy coordinates" is used, several applications can use the same LPS system and have different areas of interest for each one.

Overall, the main advantage of the absolute LPS systems is the high accuracy that they support when estimating the position of an object. However, the disadvantage is that it requires additional equipment to be carried by the located object, which although, in most cases is small and economic, doesn't help in the user-friendliness envisioned for these systems. Moreover, a main drawback is associated with the deployment costs and the operation maintenance of a second location specific infrastructure that runs in parallel to the wireless communication infrastructure.

4.2.2 Relative Positioning

Relative positioning is when location of objects/users is in terms of relation to one another (i.e., "in room 1," "near," "next to"). Many, if not most, indoor LBS applications do not need an absolute location information in terms of a xy coordinate point. Instead, a mere value of proximity ("next to") or containment ("in the room") to some object is sufficient. Each object can have its own frame of reference. For example, Active Badge LPS system provides relative position (symbolic location), which encompasses abstract ideas of where something is. For example, the person is "in the room" and therefore not in the hallway. This is shown in **Figure 4.4**.

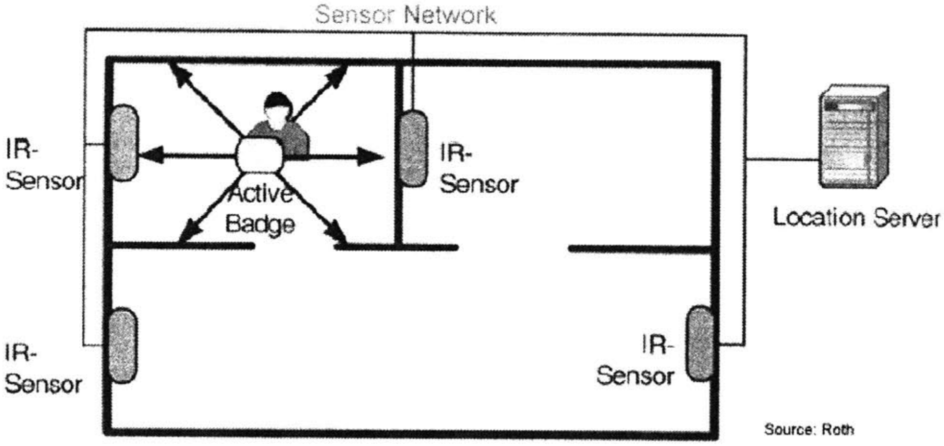
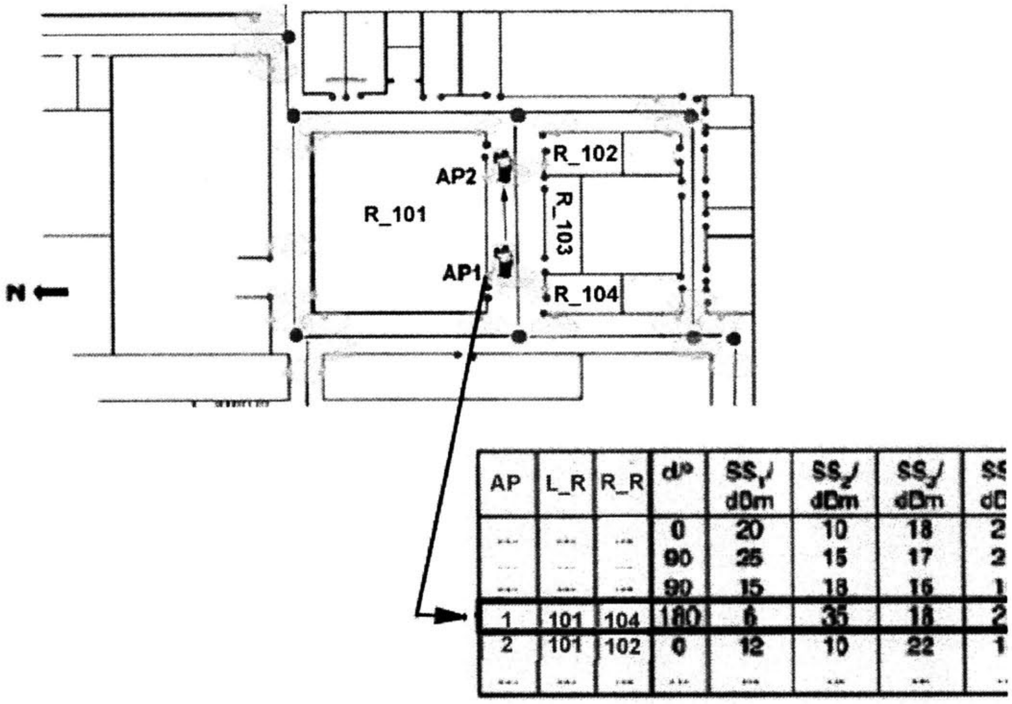


Figure 4.4 Active Badge (Image Source: Roth)

Also, point-of-sale logs, bar code scanners, and systems that monitor computer login activity are symbolic location technologies mostly based on proximity to known objects. Purely symbolic location systems typically provide only very coarse-grained physical positions. Using them often requires multiple readings or sensors to increase accuracy, such as using multiple overlapping proximity sensors to detect someone's position within a room.

Similarly to the absolute positioning with Wi-Fi, relative positioning works by means of referencing the Wi-Fi AP that the user is accessing to its location, which in this case, is relative. Wi-Fi access points (APs) transmitting RF signal augmented with their relative position of "*AP1: L_R101; R_R104*" (see **Figure 4.5**) can be used to estimate the location of the mobile host. This positioning database, which stores IDs of each AP, can potentially define more regions and be more specific (decreasing calculations to determine the user's position).



AP	L_R	R_R	d/°	SS1/ dBm	SS2/ dBm	SS3/ dBm	SS dB
...	...	...	0	20	10	18	2
...	...	...	90	25	15	17	2
...	...	...	90	15	18	16	1
1	101	104	180	6	35	18	2
2	101	102	0	12	10	22	1
...	...	...	...	...	...	...	..

Figure 4.5 Relative Positioning

Moreover, each AP has an access range/perimeter, which is represented (modeled in the symbolic database) so that a user accessing AP1 is "within" the range of that AP1. This range may or may not be defined to a specific metric distance (i.e., range of 10m). The point is that it does not need to be for such applications as the Buddy Finder because knowing that AP1 is "next to" AP2 will allow referencing (matching) that the user accessing AP1 is "next to" the user accessing AP2.

Hypothetically, if each room in a building had its own AP, and if the rules for binding to APs were strict, then the user's position could be related to the AP he is bound to. This is not practical from an infrastructure perspective, since Wi-Fi has good range and such density of APs wouldn't be economical, unless the number of users warranted it. In any case, the position is derived based on the Wi-Fi AP that the user access, of which the location (the room number) is known (i.e., Wi-Fi AP#201_1 is "in room 201"). To locate a user, then, the idea is that if '*User 1*' access '*Wi-Fi AP #201_1*,' he/she is mapped/referenced to '*Room 201*.' This is shown in **Figure 4.6**.

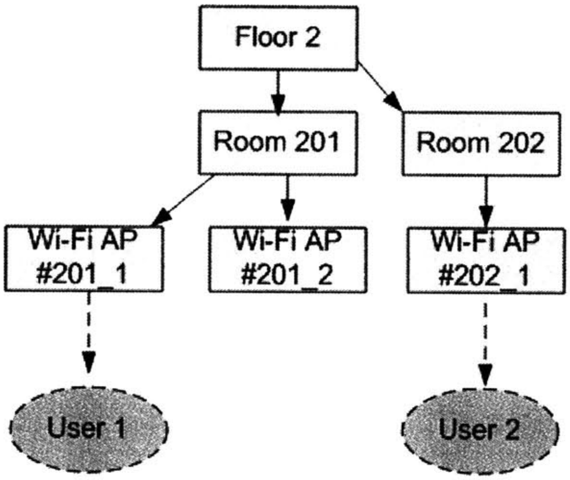


Figure 4.6 Wi-Fi Relative Positioning

Also, if a mobile device can detect signals from two APs that are known to be relatively far apart, then the device's location can be described as somewhere between these two APs. Decreasing precision in favor of logical statements such as "between object A and B" is a cell-id-base interpolation method that works with both, relative and absolute positioning.

4.2.3 Potential Challenges and Solutions

Some of the potential challenges include a Wi-Fi AP not being registered in the positioning database or the Wi-Fi signal is weak. One possible solution to the possibility of having a Wi-Fi AP not registered in the positioning database is to have this database updated by users contributing more information into the database. For example, assume a user goes to a particular room and receives beacons from three APs but only two are in the database. The third AP can then be added to the database with some high confidence that it is near the location of the other two APs. Data can also be added when an unknown AP is detected temporarily between two known APs

Also, a geographic statistical solution can be used to refine the details of the Wi-Fi positioning database as a side effect of people using their mobile device. Clearly, the data being collected by the geographic statistical technique would be much more useful if it was sent back into the infrastructure and then redistributed to all users as part of the Wi-Fi positioning database.

Absolute positioning is possible only if the exact x-y-z coordinates of the APs are known. In addition the mobile device must be in a straight line of sight for the signal to be measured correctly. Triangulation can be used to determine both absolute and relative position (i.e., intra-room positioning, such as "the SW corner of Room 101.") Hence, a person would first be symbolically located and then additional processes could geometrically refine the position – if needed.

Yes another solution is Wi-Fi location fingerprinting (patter matching), which is the matching of one set of measurements with another "reference" set contained in a database (**Figure 4.7**). In other words, a mobile device takes a "snapshot" of signals from visible APs for comparison with reference points stored in the database. The fingerprints of different locations are stored in a database and matched to measured fingerprints at the current location of a mobile user. A common signal modeling approach is to record samples of wireless signals from points in a large grid drawn to encompass either the entire floor or occupied areas of a building.

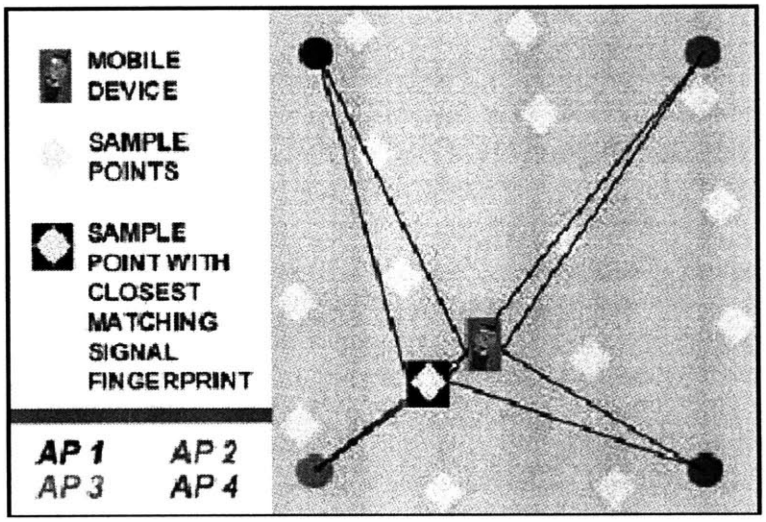


Figure 4.7: Location Fingerprinting Using a Casual Grid of Spatial Reference Points (Image Source: James Beal, Minnesota Sate University)

In spite of the additional load that databases present to a computing system, fingerprinting (a database-centric approach) has good applicability for indoor positioning, given the complexity of triangulation considering indoor wireless propagation patterns over time. For any database-oriented approach, simplicity of the reference schema is a perpetual goal. Location fingerprinting does seem well suited to the task of indoor positioning. However, it is not important to know the location of APs, as long as someone has visited each location (i.e., room) once, measured the RF signals, and saved it as a signal strength profile in a database.

The data for this local fingerprinting database can be broadcasted by having APs advertise/announce themselves similar to GPS signals. Each AP would broadcast, repeatedly, all room numbers within its signal range as well as profile information for fingerprinting. The alternative is that a user could figure out where he/she is going to be and download fingerprinting reference information from the Internet. In this case, an algorithm could determine that a user is probably closer to (or inside) Room 110. The advantage of this approach is that it is symbolic or contextual and no complicated geometry is required. Nevertheless, this is all theoretical at the moment.

4.3 “Seamless Worlds”

The significance of enabling a seamless communication and positioning handover can be seen when a company wants to expand its services into a new zone or ‘world.’ This seamless world notion can be of two types. First, there is the outdoor to indoor world seamless handover. Imagine a mobile user (standing outdoors) is trying to find an ATM. The starting location would be defined either from a cellular network measurement (or GPS or entered manually by the user (i.e., address that will be geocoded)). Upon receiving the information, the user is directed to the ATM destination which is within a shopping mall. In a cellular-to-Wi-Fi hand-off enabled world, the user would receive additional directions once inside the mall based on Wi-Fi location capabilities and would quickly find the ATM. (**Chapter 2** discusses a navigation use case in more detail.)

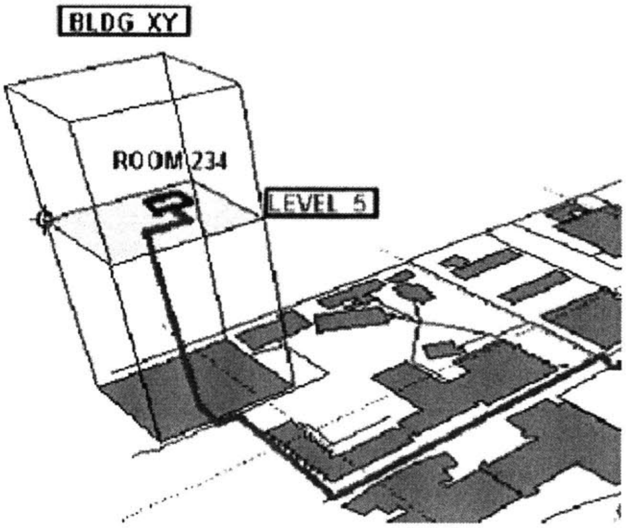


Figure 4.8: Outdoor to Indoor Seamless Handover (Image Source: Xmarc)

Second, there is the seamless handover within the sub zones (i.e., floors) of the indoor world. Each zone or floor can be equipped with a different type of positioning infrastructure, meaning, one LPS system can provide absolute positioning (xy coordinates), such as MIT Cricket or MS RADAR, while another LPS system can provide relative positioning that is purely based on Wi-Fi cell-id (here the Wi-Fi AP has no fixed coordinates but a location that is represented symbolically such as "in room 101').

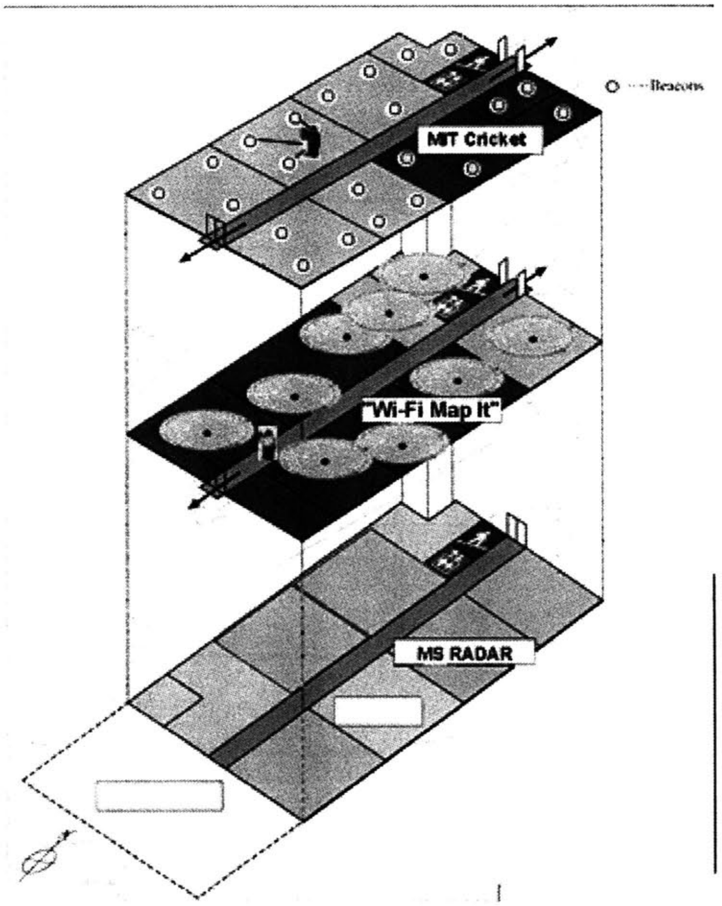


Figure 4.9 Indoor to Indoor Seamless Handover

Absolute positioning is represented using geometry in a single coordinate frame (i.e., floor map of a building). It is important to realize that there can be several zones (i.e., floors) in a building, hence, frames of reference. Potentially, these maps can be from different mapping infrastructure providers (**Chapter 5**) that would need to be in the same spatial reference system in order to enable data/map overlay. The significance of coordination among providers is explored in **Chapter 7** as part of Scenario 1.

4.3.1 Seamless Communication Handover

Seamless handover is when a hand-off from one area (cell) to another takes place without perceptible interruption of the communication connection. Two worlds can co-exist consisting of different communication networks. For discussion purposes, the two worlds are the indoor world and the outdoor world, having a Wi-Fi and GSM networks, respectively.

The cell base stations within the GSM network send out RF signals. This enables the mobile station to monitor the signal quality of available cells. Based on the measurement of the strength of those RF signals, the station decides when to switch to a new cell. The switching process is called hand-off, or handover. Moreover, the GSM network already has some basic location capability. The system knows the cell number that an active GSM user is currently located at, with an accuracy that varies from a few hundred meters in urban areas, to a few kilometers in rural areas. This method is known as the Cell of Origin (COO). Note, however, that in dense urban areas and places like shopping malls, the cell size might be much smaller, enabling higher accuracy and precision. Similarly, the Wi-Fi network has APs that are analogous to the GSM's networks base stations (cell towers) where the cell-id positioning method can be used.

There are two obvious problems with combining conventional cellular networks to Wi-Fi networks that makes cellular-to-Wi-Fi service handoff a problem. First, not all mobile devices support both cellular bearers (GSM, CDMA, iDEN, etc) and IEEE 802.11b. And second, when a user on a cellular network enters into a Wi-Fi coverage area, there is no mechanism in place to hand-off the call from the cellular network to the Wi-Fi network (even if these two networks would be provided by the same provider like T-mobile).

One possible solution is the integration of cellular and Wi-Fi networks. Taking from a real-world integration solution example, future versions of the T-Mobile software co-developed with Boingo will allow users to manage their connections between T-Mobile's Wi-Fi and GPRS networks[41]. Customers will be able to designate their network preferences or choose to get connected at the best available network speed, and the software will automatically connect them to the network of their choice.

Another possible solution is the use of location and GIS (**Figure 4.10**). Location capabilities (using Cell-ID, for example) in cellular networks can be used as a mechanism to manage Wi-Fi service handoff. Wi-Fi networks are typically confined to small geographic areas like a campus, and these areas can be mapped and represented by GIS databases as geographic zone features. Conventional cellular networks that already support location capabilities (explained earlier) are capable of tracking device locations. Combining this cellular location capability with a Wi-Fi coverage zone map gives the ability to intelligently trigger a cellular-to-Wi-Fi services handoff based on location when a user enters into a Wi-Fi zone (which can be indoors or outdoors).

---

[41] "T-Mobile Taps Boingo To Integrate Wi-Fi, GPRS": http://www.wi-fiplanet.com/news/article.php/2115671

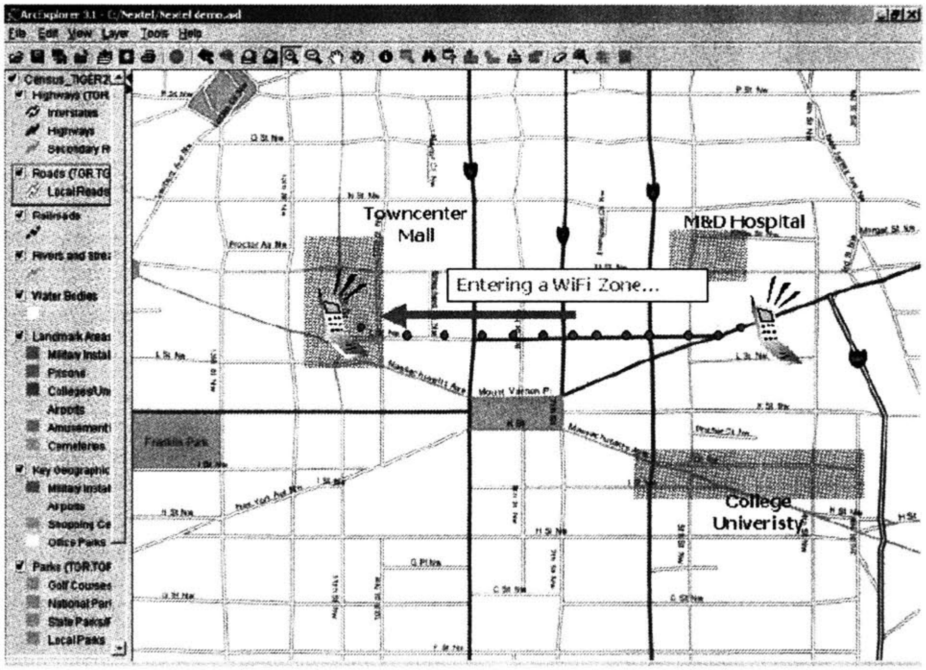


Figure 4.10: GIS Representation of a Cellular-to-Wi-Fi Services Handoff Based on Location (Image Source: ESRI)

This handoff trigger is based on the same premise of location-based presence for the purposes of zone-based alerting in cellular networks, but the zone happens to be a Wi-Fi area in this case.

4.3.2 Seamless Positioning (Location) Handover

Communication (service) continuity and hand-off issues, and roaming when combined with location adds another unique challenge – positioning handover. Seamless handover of location positioning is when a LBS service is seamless or uninterrupted from one zone (that uses GSM) to another zone (that uses Wi-Fi).

Consider the following scenario of having disparate number of local coordinate systems being utilized across the worlds. **Figure 4.11** shows that the outdoor world is using absolute positioning while the indoor world is using relative positioning. This is where the LIF MLP API (explored at the end of the chapter) comes into play by providing a standard for addressing these multiple and cross-boundary implementation challenges with respect to positioning using wireless communication networks.

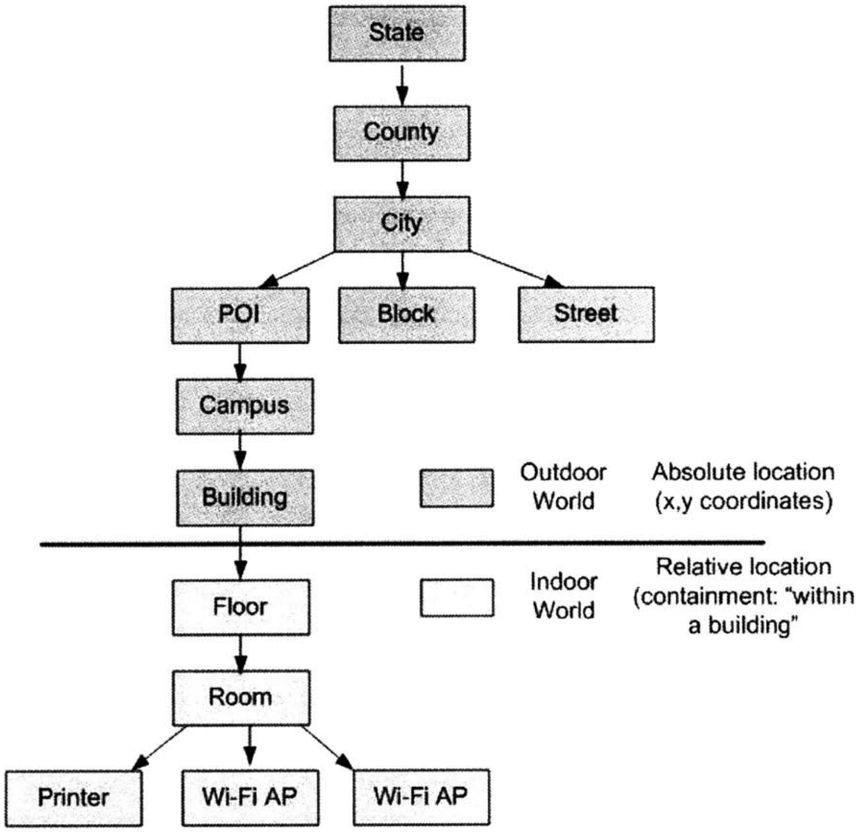


Figure 4.11: Seamless Positioning Handover: Outdoor to Indoor

In order to understand the complexity of a positioning seamless handover, one must consider the different databases and components that take part in the management of user's location data (**Chapter 6**). As explained later in the chapter, both GSM and Wi-Fi networks utilize the cell-id positioning method to locate users. Because the position of the transceiver of the wireless network is known and each transceiver has a unique identity, the user's location is inferred to be in the cell of that particular transceiver if he or she receives its identity. Wireless LAN (and Bluetooth) work differently than the mobile cellular network, but the principle at the positioning level is the same. In both, position information can be obtained from the local network.

Unfortunately, it is not possible to use the same data in the Wi-Fi network as in the mobile network if the network operator is not providing the system (since you cannot connect to the databases). Therefore, a seamless handover from a position based on the mobile telephone network to a network based on Wi-Fi network is not possible without integrating the different databases first.

One possible solution is to have a coordinating entity within the systems, like the Mobile Positioning Center (MPC). It's not good enough to receive the identity of the base station (Wi-Fi AP or the cellular tower) if the coordinates that it represents are not known. For a mobile terminal to be able to do something with the data, it needs a system to translate the address into a position. This requires the owners of the networks to create this mapping (or allow someone to do it).

As part of the solution, the mobile operators would integrate wLANs (Wi-Fi, Bluetooth) into their networks. Some manufacturers talk about combining wLANs and mobile telephone networks and using the HLR to handle the user information in wLANs as well. In that case, the MPC could give the position in a standard way (same reference system) of the user for all the different networks to which the user connects. However, there is a need for a standard for integrating positioning based on micro-location with network positioning that would allow for a centralized architecture. This is where LIF MLP API becomes useful, as explained at the end of this chapter.

Within the indoor world, different zones of positioning types can co-exist not only on different floors, but also on the same floor. **Figure 4.12** shows a user in an indoor setting such as a floor of a shopping mall. In one part (zone) of the floor, relative positioning is provided, while in another part, absolute positioning is provided.

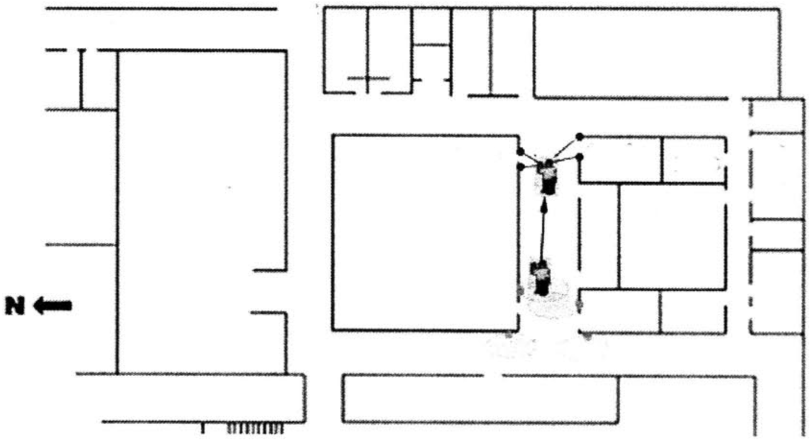


Figure 4.12 Seamless Positioning Handover: Indoor (Relative) to Indoor (Absolute)

Positioning handover affects localization of data content, which is associated with the mapping infrastructure explored in **Chapter 5**. Mapping infrastructure providers (i.e., content providers) have to figure out how to seamlessly provide location content utilizing local location model types and data formats.

4.4 Classification of Indoor Location Positioning Systems

In addition to explaining relative positioning versus absolute positioning, this chapter serves as a survey and classification for LPS systems. The relatively high number of research systems and the few commercial systems available today make indoor location sensing domain mature enough to define a system classification. The classification is broken down into the different infrastructure types and the associated characteristics; the communication infrastructure and the positioning infrastructure are discussed in this chapter, while the mapping infrastructure and the software (service) infrastructure are disused in **Chapter 5** and **Chapter 6**, respectively. Overall, this classification can help better evaluate options when choosing such a LPS system for specific indoor LBS application needs.

Table 4.1 features seven LPS systems that were reviewed in detail (the LPS manufacture was contacted to gather the information). See **Appendix A** for a detailed description of each. All of these seven LPS systems, except for two (PinPoint and UbiSense) are directly applied for indoor LBS applications. The other two are used for tracking application, but are not of a LBS service nature (there is no service by which a mobile user can request "where am I?" or "what's around me?"). These two were surveyed, nonetheless, because of their unique (precise) positioning technology. In addition, a general review of other LPS systems is included in **Appendix A** to give a sense of the magnitude of the industry. **Appendix A** also includes the definition for the positioning infrastructure's characteristics.

It is important to realize that it is a challenging problem to build an ideal indoor LPS that provides accurate and precise location at a high update rate. Different underlying sensors will give different results of accuracy and precision. Accuracy requirements for indoor LBS applications vary significantly. For example, RFID tags are considered to be proximity sensors. These can detect a tag when it passes within a relatively short distance of a sensor (usually a few feet or so). Other technologies can sense when a tag is within a room (relative positioning), but cannot identify its absolute location within the room.

Also, most available LPS systems are network-based rather than handset-based (sensor integration into the mobile device), meaning that the network calculates the position of the mobile device as opposed to a receiver inside the device calculating its own position and then a network is required for the device to notify others of its location. Network-based positioning is especially common for LPS systems that are typically based on small RF or IR cells. Handset-based positioning may conflict with resource limitations on the mobile unit. As a matter of fact, all of the LPS that were part of the detailed surveyed are networked-based.

Network-based positioning allows communication infrastructure providers to own the data and to ensure that its use generates revenue for them. A key benefit of network-based solutions is that all handsets can utilize the positioning technology without modification. But network solutions are more expensive than handset-based solutions, since each base station must be upgraded. Alternatively, handset-based solutions are more accurate and less expensive to deploy, since the handset expense is passed on to the subscriber as a one-time charge. However, handset-based solutions are only available to subscribers who purchase the new handsets.

Another general characteristic or requirement of LPS systems is the reasonably high update rates, happening in real-time. In general, one or more location updates a second are required for many of LBS applications. An important aspect of a practical LPS is being able to intelligently vary update rates of individual tags in a dynamic fashion. One of the keys to power management is to intelligently vary the update rates of individual objects. A fast moving object may need to have its location updated multiple times a second, whereas a stationary one does not. Certain objects may need more frequent updates than others.

GIS systems are database-centric systems used to store, retrieve, and analyze spatially referenced data. This functionality can be extended to track the location of mobile objects. Furthermore, GIS is an essential element of the LBS value-chain to provide additional location data management and functionality. Also, some of the techniques for location modeling and indexing developed for GIS are also applicable in the wider context of general LBS services. See **Chapter 6** for more on GIS.

<< SYSTEM >> PROPERTIES	(1) Active Badge	(2) Active Bat	(3) RADAR (Microsoft)	(4) Cricket (MIT)
>> Communication Infrastructure	Cellular IR	Cellular RF	Cellular RF? 802.11 (Wi-Fi)	Cellular RF
Network Type	Network-based	Network-based	Network-based	Network-based
Heterogeneity of networks	Yes	Yes	No (?)	Yes
>> Positioning Infrastructure	Needs extra deployment…	Needs extra deployment…	Leverages existing comm. infra…	Needs extra deployment…
Positioning Type	Relative	Absolute	Absolute	Absolute
Positioning Method(s)	Diffuse IR cellular proximity	Cell-id (triangulation; TOA to calculate the position); RF and ultrasound (ultrasonic grid)	Cell-id (triangulation for RF signal & scene analysis for ultrasound signal)	Proximity lateration; TOF (distance readings); RF and ultrasound
Orientation	No	Yes	Yes	Yes
Accuracy	Room size	9 cm (95%)	2-3 meters (3-4m) (50%)	4 x 4 ft regions, 100% (few inches possible)
Scale	1 base per room, badge per base per 10sec	1 base per 10sq m, 25 computation per room per sec	3 bases per floor	1 beckon per 16sq ft.
Recognition	Yes	Yes	Yes	No
LLC			No	Yes
User Privacy	No	No	Yes	Yes
Decentralization	No	No	No (Centralized RF signal db)	Yes
Cost	High	High	Low	High

Table 4.1: Featured LPS Systems (1)

<< SYSTEM >>	**(5)** **MobileShadow** **(SafeSoftware)**
PROPERTIES	
>> Communication Infrastructure	RF (802.11b)
Network Type	Network-based
Heterogeneity of networks	Yes (could be easily coupled with Bluetooth or RFID tags)
>> Positioning Infrastructure	
Positioning Type	
Positioning Method(s)	Cell id
Orientation	Yes (3d)
Accuracy	(Depends on cell size)
Scale	As large as 802.11b scales, the 802.11 standard limits the number of users
Recognition	Yes, recognizes users and provides proactive services just for this one authorized user
LLC	
User Privacy	Against other users: yes, for the system administrators: no
Decentralization	Yes (the key aspect)
Cost	802.11b nodes + mini computer per location, hierarchical system: several cells can be combined to larger areas
>> Mapping Infrastructure	
Location Model Type	Symbolic position
Modeling language	Proprietary (could use XML)
Data format	Text based
Content	Individual services mainly communicating via html
>> Software Infrastructure	Open architecture, mobile agent based -> each user can write his/her own services
Applications / Services	Proactive
Overall limitations	Cell id, system security because of open architecture, drawbacks that result from using mobile agent technology

<< SYSTEM >>	**(6)** **UbiTags (UbiSense)**
PROPERTIES	
>> Communication Infrastructure	RF (bi-directional)
Network Type	Network-based
Heterogeneity of networks	Yes
>> Positioning Infrastructure	Ultra Wideband (UWB) Radio; UbiTags only transmit UWB, sensors only receive UWB
Positioning Type	TOA + AOA
Positioning Method(s)	Absolute
Orientation	Yes (Implied orientations derived by using motion vector, multiple tags, or sensor shadowing information)
Accuracy	15cm, 95% of the time
Scale	100 – 1000m2 per cell, depending on environment, typically 4 sensor units per cell, 1 ubitag per object (person)
Recognition	Yes
LLC	No
User Privacy	No
Decentralization	No
Cost	Low (due to leveraging existing communication network, but still requires network of dedicated sensors.
>> Mapping Infrastructure	
Location Model	Geometric
Modeling language	
Data format	3D vector + Quaternion or Euler angles + out-of-band accuracy data. Also inter-object spatial relationships.
Content	Arbitrary additional content depending on application environment.
>> Software Infrastructure	Open architecture. Unlimited numbers of simultaneous applications. Tools and APIs for specification and monitoring of geometric relationships.
Applications / Services	Proactive

Table 4.1: Featured LPS Systems (2)

4.5 Overview of the Communication Infrastructure & Positioning Infrastructure

4.5.1 Cellular Mobile Networks

In principle, all mobile communication networks work in the same way. The user connects to a base station that handles RF signal (based on the GSM, CDMA, WCDMA, TDMA, PDC, or PHS radio standards) that is connected to a network. It connects the user to the network either through setting up as ISDN connection or to the Internet. The primary purpose is to set up communication channels between mobile parties and to help with the setup of mobile-terminating calls.

The wider category of mobility management also includes authentication and security functions. Location information is only accessible to the network operator via the signaling networking (typically Signaling System No. 7or SS7[42]). SS7 functions exist to handle the positioning of a terminal, encoded in the internal system. Developers can access those functions using the LIF MLP API as discussed later in the chapter.

GSM already has some rudimentary location capability. The system knows the cell number that an active GSM user is currently located at, with an accuracy that varies from a few hundred meters in urban areas, to a few kilometers in rural areas. This method is known as the Cell of Origin (COO). The characteristics of location positioning system - resolution, accuracy, and system responsiveness are very acceptable using COO technology for the early adopters of the positioning infrastructure. COO requires no modification to the handset or networks thus permitting its use as the positioning for existing subscribers but is less accurate than the other methods employed.

Moreover, while using this existing communication infrastructure has been found to be promising in outdoor environments for coarse-grained location services (i.e., E-911), its immediate applicability in indoor environments for more precise indoor LBS services is limited. However, in urban areas the cell size of the GSM network might be small enough to allow for indoor LBS applications.

4.5.2 PCS Location Directories

It is anticipated that future personal communication services (PCS) will have a much smaller cell sizes and much increased number of customers than the current GSM implementations. It is argued that this will lead to query and update volumes several magnitudes higher than in GSM systems. As a result, there has been a lot of research into the efficient and scalable tracking of mobile users in future PCS networks.

Such systems contain two basic operations: “move” (location update) and “find” (location query). Typically, proposed solutions use a multi-level hierarchy of location servers. Each location server node has a well-defined network coverage area. Location updates are triggered by

---

[42] SS7 is a special protocol developed for the telephony applications or through IP. IP-based mobile networks were not deployed at the time of writing, because the standards have not been finalized yet.

the leaf nodes, and are propagated through the directory following a well-defined algorithm. In contrast, queries can generally be directed to nodes at any level in the hierarchy. This often leads to recursive query patterns.

The basic mechanisms employed to improve response time and reduce network traffic are data replication and forwarding pointers. Data replication reduces query latency and query traffic but increases update traffic. Also, consistency control is normally achieved by simple timeouts. Since efficient queries and updates need to make assumptions about parameters of user mobility (such as the call-to-mobility ratio), sometimes systems can adapt dynamically to changing user characteristics. The subscriber profile would be used to store such mobility parameters. Unlike GSM, many approaches do not rely on an HLR for location queries. The location server hierarchy subsumes the HLR's location management function.

4.5.3 Wireless Local Area Network (wLAN)

wLAN or "Wi-Fi" (Wireless Fidelity) networks such as 802.11 and Bluetooth, are characterized by a number of base stations (access points) placed throughout the networked environment and connected to the traditional wired LAN. Each station has a range of roughly 300m in open space, and interference between different stations is dealt with by using different channels and by a CSMA/CA access protocol. Devices are connected by Wi-Fi cards that typically communicate with the access point having the strongest signal.

Access points (APs) transmitting RF signal can be used to estimate the location of the mobile host. The strengths of the RF signals arriving from more APs are related to the position of the mobile terminal and can be used to derive the location of the user. The accuracy of the system might be limited by the (possibly large) cell size. Roaming between APs is supported, and Wi-Fi networks can be extended to create "clouds of connectivity" inside the so-called hotspot, i.e. locations with high connection frequency such as office buildings.

In a wireless cellular network like GSM, there is a user identity built into the mobile device (or station) that is unique. In 802.11 networks, it is derived in the same way as in the Ethernet, from a hierarchic number series. This unique id is what is used when locating the mobile device. A mobile phone gets a number in a similar way, which is allocated by the International Telecommunications Union (ITU) through a process where it is actually allocated by other standardized bodies.

4.6 Communication Infrastructure Characteristics

A network environment is about the availability of a network connection allowing information exchange between devices. In general, there are three types of networks (environments) to managing location information: (1) network-based; (2) ad-hoc-based; (3) hybrid-based (combination of both).

See **Appendix B** for more on the communication infrastructure's characteristics.

4.7 Positioning Infrastructure Characteristics

Indoor LPS systems rely on the following positioning methods (these can be used on their own or jointly):

- Triangulation
    - lateration (TOF, attenuation)
    - angulation
- Scene analysis
    - physical contact through pressure sensors
    - Monitoring (Wi-Fi APs for determining when an object is in their range)
    - Observing (automatic ID systems)
- Proximity

See **Appendix A** for a description of each, in addition to information about each of the featured LPS systems as well as others. The Appendix also includes explanations of positing methods and descriptions for other sensor types used by these other LPS systems.

4.8 Use of Open Interoperability Standards

There are two (accepted) standards that promise to expedite integration within the communication and positioning infrastructure and stimulate application development within the software (service) infrastructure:

1) the LIF's MLP API (for the communication and positioning infrastructure)
2) the Open GIS Consortium (OGC) OpenLS API for spatial processing (for the software (services) infrastructure)

Communication infrastructure providers can benefit from standardized location positioning protocols like the LIF MLP API. Having the indoor and outdoor world (or subsets of either world) using the same standard for modeling / encrypting user's positions would enable seamless positioning handover from one region/zone to the other without the need for data format conversations and reference system transformations.

Moreover, the LIF MLP API enables integrating positioning based on micro-location with network positioning that allows for a centralized architecture see (**Chapter 6**) allowing a one-stop shopping environments for developers.

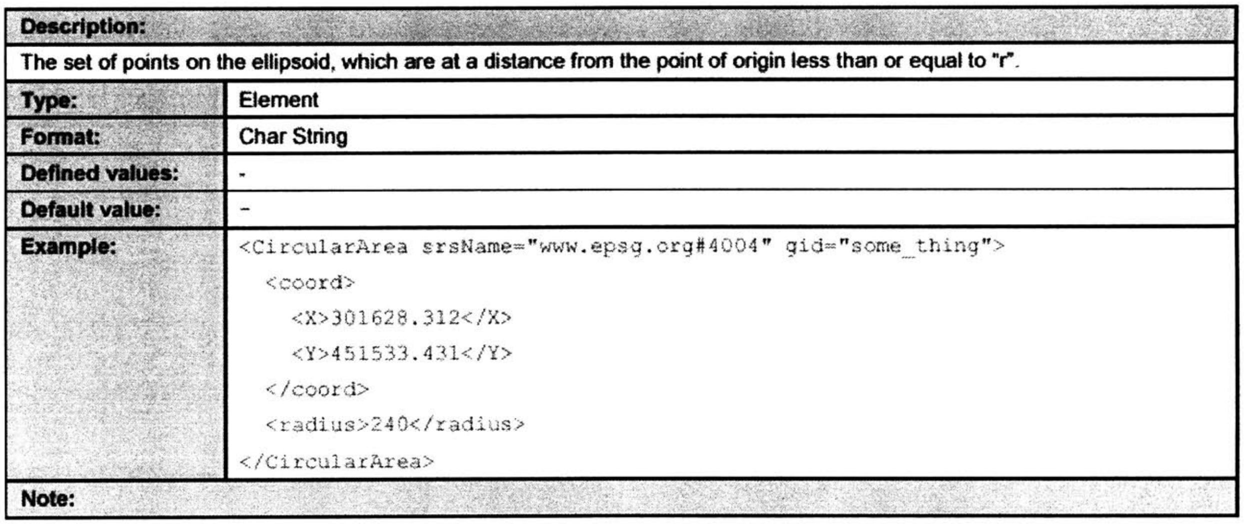
4.8.1 LIF Mobile Location Protocol (MLP)

LIF has developed the Mobile Location Protocol (MLP) standard protocol for utilization by communication infrastructure providers and the software (services) infrastructure providers (i.e., mobile application developers / the vendors). **Chapter 6** discusses how the LIF MLP API is used to seamlessly integrate location from the communication network with the software components. In this chapter, we discuss how the LIF API can be used to model user's position.

The LIF API defines a number of geometric shapes used in defining a geographical position. When requesting and reporting a position, the result can be a point with xy coordinates. To assess the applicability of the LIF API for the indoor world, it is necessary to apply it to both absolute and relative positioning. Indoor LPS systems operating on absolute positioning like MIT Cricket and RADAR can supply xy coordinate location for which the LIF MLP API can be used to model position location. However, LPS such as Active Badge give a symbolic location ("in room 101") and this is where the LIF MLP API might be lacking, in terms of modeling relative positioning.

To explore the matter of modeling relative positioning using the LIF MLP API, consider the following. Since it is rare to have a precise position like a xy point coordinate, the LIF MLP API can describe position in terms of inaccuracy as a circle or a polygon (or some other shape) with a radius that describes the inaccuracy. Specific elements of the LIF MLP API that could be considered to model this include CircularArea, CircularArcArea, Polygon (**Figure 13**). The CircularArea element can be used to define the Wi-Fi AP's range within which the user is positioned. These elements represent geometric shapes that have absolute positions (e.g., an arc is made up of points that have xy coordinates), and, hence, are not suited to model symbolic locations like "user 1 is in room 101" as he/she is accessing the Wi-Fi access point (AP) that is located in room 101. It would seem that symbolic locations like "in the room" or "in the hallway" would need to be defined for the LIF MLP standard to define relative positions in a common way.

The LIF MLP API can also be used to set quality requirements on the position information by using the LEV_CONF attribute, which indicates the probability in percent that the mobile station is located in the position area that is returned. It is a percentage value associated with the accuracy. If a location positioning system can determine that a user is in a circle sector that is long and narrow, and by measuring the circle arc in which the user is located and finding that it is narrow and broad, or determining that the user is in a cell, there is a tradeoff between the size of the inaccuracy area and the accuracy required. It is not very useful to know whether a user is within a certain cell if it is large, but it will help to know whether the user is somewhere in a circle sector.



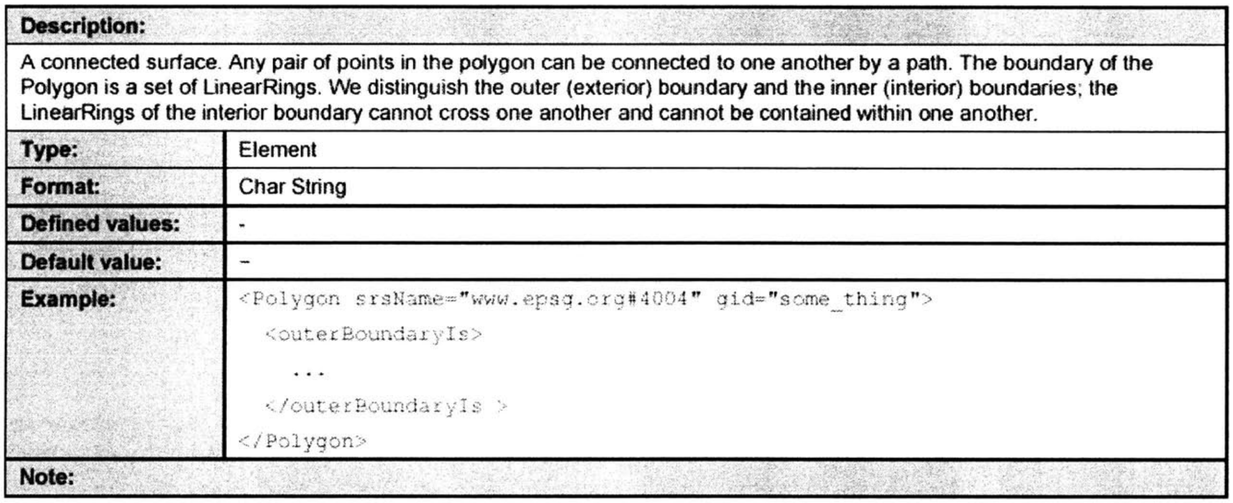
Description:	
The set of points on the ellipsoid, which are at a distance from the point of origin less than or equal to "r".	
Type:	Element
Format:	Char String
Defined values:	-
Default value:	-

Example:

```
<CircularArea srsName="www.epsg.org#4004" gid="some_thing">
  <coord>
    <X>301628.312</X>
    <Y>451533.431</Y>
  </coord>
  <radius>240</radius>
</CircularArea>
```

Note:

Figure 4.13a: LIF MLP API "CircularArea" Element



Description:

A connected surface. Any pair of points in the polygon can be connected to one another by a path. The boundary of the Polygon is a set of LinearRings. We distinguish the outer (exterior) boundary and the inner (interior) boundaries; the LinearRings of the interior boundary cannot cross one another and cannot be contained within one another.

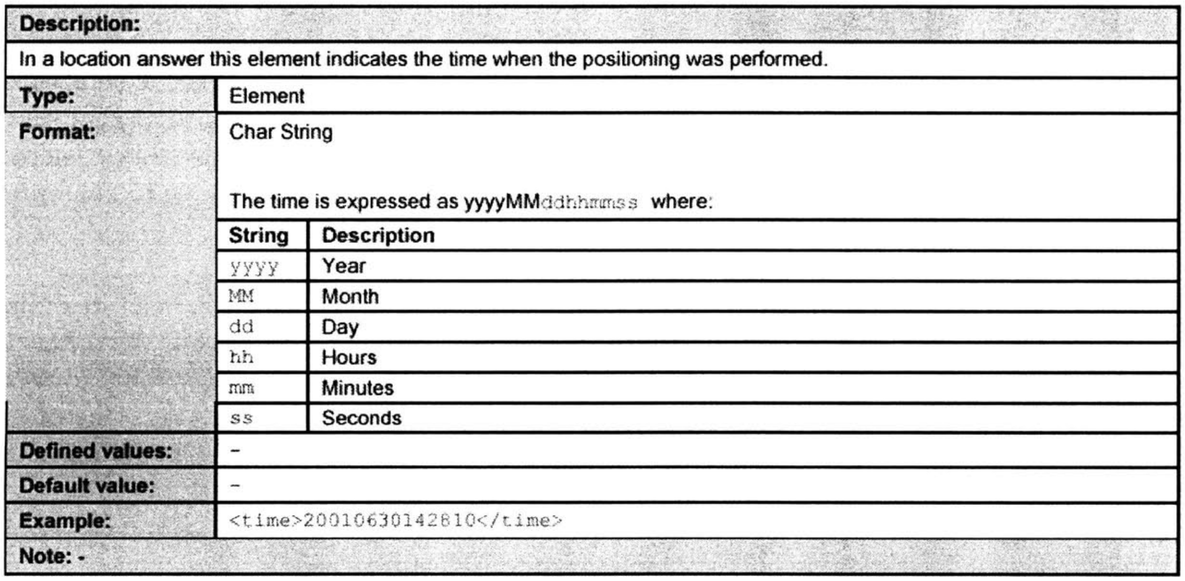
Type:	Element
Format:	Char String
Defined values:	-
Default value:	-
Example:	

```
<Polygon srsName="www.epsg.org#4004" gid="some_thing">
  <outerBoundaryIs>
    ...
  </outerBoundaryIs >
</Polygon>
```

Note:

Figure 4.13b: LIF MLP API "Polygon" Element

The "Time" element can be used in calculating position based on the time of arrival (TOA) geo-location method. The location can be one of four different types: current, last known, current or last known, or initial emergency call location.



Description:

In a location answer this element indicates the time when the positioning was performed.

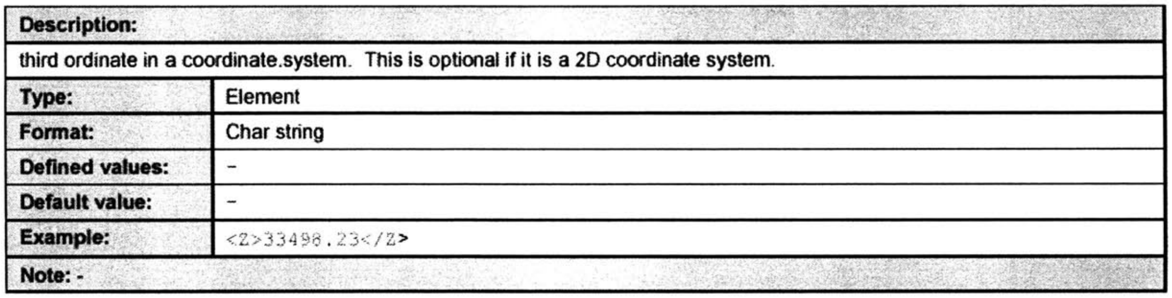
Type:	Element
Format:	Char String The time is expressed as yyyyMMddhhmmss where:

String	Description
yyyy	Year
MM	Month
dd	Day
hh	Hours
mm	Minutes
ss	Seconds

Defined values:	-
Default value:	-
Example:	`<time>20010630142810</time>`
Note: -	

Figure 4.13c: LIF MLP API "Time" Element

In addition to the "X" and "Y" elements, LIF MLP API also includes the "Z" element for 2D.



Description:	
third ordinate in a coordinate.system. This is optional if it is a 2D coordinate system.	
Type:	Element
Format:	Char string
Defined values:	-
Default value:	-
Example:	`<Z>33498.23</Z>`
Note: -	

Figure 4.13d: LIF MLP API "Z" Element

4.8.2 OGC Sensor Web

As already mentioned, LPS systems include many types of sensors including IR proximity badges, passive RFID tags, ultrasound ranging tags, etc. These sensors typically have proprietary means for management, control and access of data. Each sensor driver collects and classifies the data produced into measurements of type distance, angle, proximity, or position. In addition, each measurement has an uncertainty model derived from the physical characteristics of the sensor and the environment. This data is then stored in the location models (**Chapter 5**). Also, currently, these systems are not accessible through online services, which would allow for dynamic updated of information within a distributed computing environment.

A standard solution is sought for online access to sensor systems. In particular, a solution is desired for any applications that have access to the Web to access sensor assets through a set of Web services (**Chapter 6**). With improvements in sensor, communications and Internet technologies, it is feasible to construct these capabilities in the form of a 'Sensor Web' consisting of loosely coupled processing components.

For these reasons, OGC is promoting the concept of the Sensor Web, which is a set of online Web services that provide access to sensor assets are sought. These services provide the means to determine collection feasibility (i.e., allows to determine if sensor assets are available to meet specific needs), and to collect observations from available sensors in a standard way. One of the capabilities of Sensor Web is the means to model sensors and encode information about sensors (SensorML).

4.9 Conclusions

In this chapter we show that positioning can be either relative or absolute. Most LPS systems are of the absolute positioning type. This seems to be because with "xy coordinates" there is more flexibility and abstraction for application development than with relative positioning (using symbolic representation). This in turn, is not limiting the logic of the application.

The absolute/relative is drawn out with a geometric or symbolic distinction, point being that Euclidian may or may not use a standard world coordinate system and symbolic requires some database structure for topological relationships. The positioning handoff requires knowing the Euclidian position of some nodes in the topology to within some tolerance. A seamless handoff is hard with topology since other worlds won't have access to (and won't understand) the topology database without some kind of a common (standardized) referencing/mapping between the two worlds.

Overall, it is enough to have a user to be first positioned relatively ("in room 101") and then have additional geometrical processes (i.e., triangulation) to refine the position to absolute terms, if needed. Since this is a less accurate system it is valid to have a higher level of abstraction where a symbolic language (**Chapter 5**) can be used. Both types of LPS systems could use the positioning information to feed a bigger symbolic system. Usefulness of each depends upon scale/resolution and match with technology. For example, Wi-Fi matches crude building/floor/room positioning and can facilitate Euclidian/topology handoff at that crude scale but can't easily handle the use cases from **Chapter 2** that need accurate positioning, even to "*within a room*."

Given that both symbolic and geometrical systems could co-exist that would be integrated into a bigger system based on a symbolic language. This is a good reason to have a standardized language for relative positioning. Developing a world-wide standard for this symbolic language would not be difficult but it would need something similar to the Domain Name Service (DNS) to publish all used tags and avoid conflicts.

In most cases where a simpler approach (i.e., a single tag and sensor infrastructure instead of multiple ones) might suffice and be significantly cheaper for some basic indoor LBS application, there are additional LBS applications of interest that can only be implemented with a more precise LPS. Hence, it is important to consider whether the simpler approach will suffice for all indoor LBS applications that may be needed in the long term. If not, it makes sense to implement a positioning infrastructure that will handle all LBS applications from the simplest to the more sophisticated.

Still, today's indoor LPS systems require additional infrastructure investment, often significant. But, we also think that an approach of storing location profiles in mobile friendly databases and using Room numbers as the nodes in a symbolic location model could minimize the need for positioning infrastructure development. The cost of commercial deployments of location-aware technology can be reduced by appropriate design and quantified before the system is installed, and a properly-designed infrastructure will have minimal maintenance requirements once installed. Furthermore, fixed infrastructure lets the vendor give predictions and guarantees about system performance, giving purchasers' confidence that applications will be robust.

Also, indoor LPS systems currently focus either on a particular application or on a specific location positioning (sensor) technology. The relative immaturity of the indoor LBS market suggests that 'full-system' sales (including sensor hardware, data management middleware, and, perhaps, application software) will predominate for some time, until the understanding and language required to define appropriate interfaces at each level of the system have been developed.

In fact, it may be that even when suitable interfaces have been identified, the limitations of sensor technology will make it difficult for middleware to completely abstract away the properties of supporting sensor systems, leading to a more complex relationship between players at different levels of the system in the location-awareness space. However, as seen with OGC's Sensor Web and Web Services (**Chapter 6**), applications are abstracted away from the underlying sensor and positioning technology.

Also, it is foreseen that a combination of at least two location sensing technologies (e.g., RF and ultrasound) will take place, since one technology poses too many limitations to be accurate by itself. The flexibility of choosing the underlying positioning sensors will become an increasingly vital attribute of the supporting location platforms (**Chapter 6**). One way about this is to work with communication infrastructure providers in designing an overlay network to abstract these specifications from an application point of view.

Overall, the LBS positioning infrastructure will most likely be faced with having to support multiple, disparate location positioning technologies, overabundance of positioning/location data and services and multiple data transport protocols. As mentioned in **Chapter 3**, this havoc signifies the need for open interoperability standards to make the LBS industry more homogenous enabling faster deployment of LBS services. In this chapter we presented how the LIF MLP API can make integration easier among the communication and positioning infrastructures. Next, in **Chapter 5**, we present the mapping infrastructure and the role of OGC's GML as a standard data format and representation of positioning and location/content information.

CHAPTER 5

THE MAPPING INFRASTRUCTURE

5.1 Introduction

As reviewed in **Chapter 4**, there are many indoor location positioning technologies and local positioning systems (LPS systems) that make indoor positioning and tracking possible. Although developing this positioning infrastructure that will allow for fine-grained location sensing/positioning is an essential contribution, making indoor LBS services possible is also ultimately about the mapping infrastructure (i.e., data integration, location models), which is discussed in this chapter, and the software (service) infrastructure, discussed in **Chapter 6**. Existing system may provide very accurate location information of a particular type however, it alone is almost certainly not the ultimate solution in the problem space/domain. Any robust and scalable location sensing architecture needs to develop a data model to characterize the notion of location as described by a heterogeneous mix of sensors.

For any LBS service to give meaning to the mobile client's location, it must fit that location into a 'world model'. This world model might be a geographic map based on Euclidian geometry (such as might be produced by a GIS system), or it might be a more symbolic representation of space such as a network graph describing the connectivity relationships among physical objects of interest. Once again, one size does not fit all applications and we will describe the predominant approaches that have been adopted to date.

Indoor LBS services require micro-detailed geo-referencing or micro-geography to satisfy the growing needs of users for the indoor experience. It is not enough to geo-reference a building if the position of users and other objects inside the building is also relevant. As a result, traditional geographic information is not detailed enough to satisfy the needs of users in an indoor setting. With more technologically advanced mobile devices and indoor positioning systems, new capabilities will help the development of applications and services for the indoor setting. These will go beyond traditional way finding in shopping centers, airports, schools, museums and other public spaces that have not been in the realm of traditional geography. They will include querying objects, community building (relating people, events, and objects based on location), and entertainment (location-based games).

Chapter 4 discussed relative positioning versus absolute positioning and the issues/challenges associated with "seamless worlds" – communication and positioning (location) seamless handovers. In order to achieve "seamless worlds" the required information such as a relative position and/or an absolute position must somehow be stored, modeled, and mapped. This is the role of the mapping infrastructure, where location (or "space" or "world) models are essential. It is also about the content/context data that can be associated with that location data for specific LBS services.

Also, part of the mapping infrastructure is a modeling language for the exchange of all this information. LPS systems structure information according to the position of the objects, but since their different properties (i.e., sensor type) lead to different ways to express and model them, there is a technical challenge with respect to data integration and sensor fusion. Hence, a common data modeling language is needed in order to be able to integrate sensor data from different systems. Moreover, building a new indoor LBS application is relatively easy, and gathering and organizing the data is not so hard. However, if data from one application needs to be integrated with data from another application, a common data modeling language is essential.

5.2 The Significance of the Mapping Infrastructure

The mapping infrastructure is about storing and managing position information data as well as content data associated with the location. While the most basic LBS applications can answer the standard question, "*Where am I*?" to varying degrees of accuracy, there exists a need to frame '*where*' in the context of a modeled environment (or world) in order to move beyond simple inferences of position to a better understanding of what '*where*' relates to contextually. Hence, a "world" data model is about contextual information, which includes location.

To explain the significance of the mapping infrastructure, we consider the Buddy Finder and Product Finder applications for discussion purposes. As explored in the use cases for these applications (**Chapter 2**) the Buddy Finder application will function based on relative positioning where proximity of one user in relation to another user is enough to cause a location-based trigger that will notify both users that they are in the same area (i.e., hallway). The Product Finder application, on the other hand, would need more precise positioning that is not based on proximity or containment, but on the exact xy coordinates (absolute positioning) of a product in a store isle.

The argument (as explored in the use case, **Chapter 2**) can go both ways regarding the need for absolute positioning (and navigation) for such application as the Product Finder that will help the user find the product. Relative positioning could be good enough to locate a user in relation to the aisles in the stores and notify him/her that this is the aisle for the product that he/she is looking for, adding a symbolic description for the location of the product and notifying the user with "post 3, 2nd shelf from the bottom." In any case, absolute positioning is seen valuable to handicapped customers (those that are either blind or on wheelchairs), and deserves further exploration in terms of mapping / modeling location and navigation.

An example of a world model is from the Sentient Computing Project[43] by AT&T Cambridge, which has developed the Active Badge and the Active Bat LPS systems (**Chapter 4**). This project explored what could users do if computer programs could see a model of the world. By acting within the world, users would be interacting with programs via the world model as through a user interface. While humans can observe and act on the environment directly, application programs observe and act on the environment via the world model, which is kept up to date using sensors and provides an interface to various actuators. If the expressions used by the model are natural enough, then humans can interpret their perceptions of the world in terms

---

[43] Sentient Computing Project. Website, 2003: http://www.uk.research.att.com/spirit/

of the model, and it appears to them as though they and the computer programs are sharing a perception of the real world.

Another example of a world model is the Augmented World Model (AWM) from Nexus, which provides the whole location context for context aware applications, for both the indoor and outdoor world. Two main components responsible for the main aspects of the world model are the spatial data and the position information of mobile objects, which are discussed in this chapter. This includes the representations of geometric objects (static and mobile) using geographic coordinates, and also virtual objects (i.e., virtual billboards, virtual Post-its or virtual kiosks) with which the real world is augmented. Virtual objects provide, among other things, links to external information spaces like the Web.

The following is a discussion about mapping out, or modeling, the indoor world using various types of location maps and data models.

5.3 Mapping Out the Indoor World: Location Data Models

To demonstrate mapping out the indoor world, we consider a shopping mall (or a university campus) as an example, which relates to the Buddy Finder and Product Finder LBS applications. The level of mapping detail of the underlying location model determines what functionality can be provided by the application. For example, if the application is used outdoors, it is sufficient to return places at the granularity of buildings, whereas indoors, the rooms of the building the user is currently in are appropriate. In order to get the interconnections between places, it is necessary to have relations between the objects involved. Some relations are modeled implicitly when the geometry is modeled, e.g. which rooms are next to each other. However, this may not be sufficient, if the geometric model lacks information about doors between rooms. In this case, this relation has to be explicitly modeled. Moreover, only the interesting parts of the shopping mall could be mapped geometrically (i.e., ATM locations) while others (for example corridors) could have a pure topological representation (Tomatis et. al., 2003)[44].

Also, different indoor LBS applications deal with different scales. For example, both, the Buddy Finder and the Product Finder can operate on a building scale (i.e., "*notify me when Bob is the shopping mall...*" and "*show me the location of the store that has the product that I am looking for...*"). In addition, both applications could require room-scale location model (i.e., "*notify me when Bob is in the same store/room...*" and "*show me the aisle where I can find the product...").* As a result, it is important to make sure that the underlying location model is suitable for the aimed functions. In addition, having the appropriate scale and federation of data for would enable faster querying and data processing.

---

[44] Tomatis, N .,Nourbakhsh, I . and Siegwart, R. (2003). Hybrid simultaneous localization and map building: a natural integration of topological and metric, Robotics and Autonomous Systems 44 (1): 3-4.

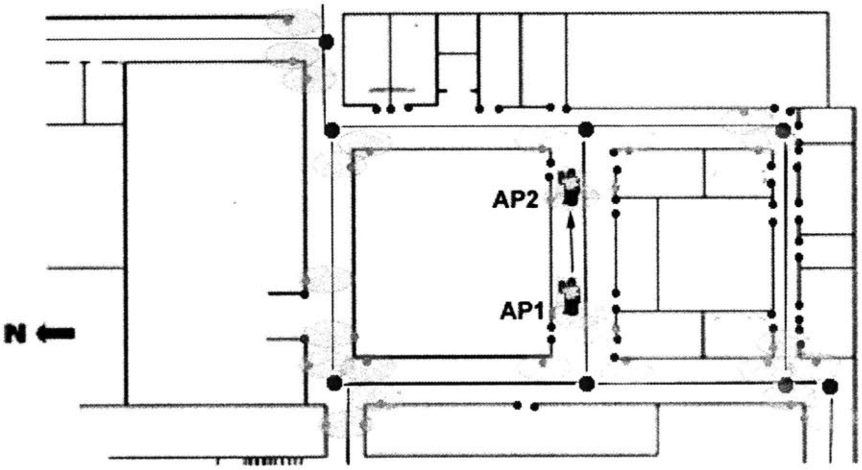


Figure 5.1 Mapping the Indoor World

Different types of location models exist that provide an abstraction between users/devices and the raw data provided by various location sensing technologies (**Chapter 4**). Numerous location models have been defined in different application domains (Bauer et. al., 2001)[45] and (Brumitt et. al., 2001)[46]. They implicitly underlie the mapping infrastructure, yet they are rarely set in a proper theoretical framework by going back to the basics of what a location (space) can be in pure mathematics, and, more importantly, symbolically (semantics). Though these models are purely abstract, they are used in association with a particular location-sensing technology, from which they retain only relevant characteristics that can be mapped to a corresponding notion of location.

In general, location models can be classified into four types:

1. *Symbolic* - describes location and space in terms of names and abstractions (i.e., hierarchies). Unlike the geometric model, humans and computational devices can understand this model better. However, they lack the precision of geometric models in terms of metrics for location and distance. (Types: spatial model graph, topological.)
2. *Topological (or structural)* – location entities as subsets or neighborhoods of space but also their structural relationships (i.e., connectivity).
3. *Geometric* - allows points, areas (2-D) and volumes (3-D) to be modeled; however a point in geometric space has no relationship to what it points to. The resolution of this model is as fine as the units of measurement used.
4. *Hybrid* - represents a logical step forward in combining the advantages of the geometric, topological, and symbolic model types in order to overcome their respective disadvantages. As a consequence, the hybrid model is more complex, requiring greater amounts of data.

For scalability and abstraction, locations are typically organized hierarchically in both the geometric and symbolic models. For example, the ActiveMap service from ArialView

---

[45] Bauer, M., Becker, C., and Rothermel, K. *Location models from the perspective of context-aware applications and mobile ad hoc networks.* In Workshop on Location Modeling for Ubiquitous Computing (2001).

[46] Brumitt, B., and Shafer, S.. *Topological world modeling using semantic spaces.* In Workshop Proceedings of Ubicomp 2001.

Awareness System uses a symbolic model with a location containment hierarchy. In their geometric models, Nelson and Ward use an R-tree index and a Quad-tree[47] index respectively to facilitate location searching and updating. The EasyLiving project[48] from Microsoft Research uses a geometric model with flat layout that works well for small scales (i.e., a room). Trying to take advantage of both symbolic and geometric models, Leonhardt proposes a hybrid model, in which a location contains both a symbolic name and geometric coordinates[49] . The symbolic name and geometric coordinates can convert to each other via pre-defined predicates. Such a combined model shields the details of underlying sensors and can support applications that need or could use both symbolic and geometric location information. Containment and intersection are the most frequently seen and probably most powerful relationships in a location model.

Symbolic information and attributes might be also applied to an intentional naming system (MIT's INS by Balakrishnan[50],), or diffusion (Estrin[51]). Both positioning, and ins/diffusion may be applied to the same end of resource/service discovery (**Chapter 6**).

5.3.1 Symbolic Maps and Location Data Models

The following is a discussion of mapping out or modeling the indoor world using symbolic location modeling. In this type of model, spatial location may be defined implicitly rather than explicitly, by reference to more or less abstract concepts relevant to a given universe of discourse, i.e. a semantic frame of reference.

A location entity, such as a shopping mall or university campus is decomposed into several intersected sub-spaces: building 1, building 2, building 3, etc. Each of these buildings is divided into smaller composing sub-spaces (floor 1, floor 2, floor 3, etc.), until enough level-of-detail is reached to reference / map objects (i.e., a printer, Wi-Fi AP, product, etc.) in a space, which in this case is a room, a store, or a shelf in a store. A portion of this decomposition or hierarchy is shown in **Figure 5.2**.

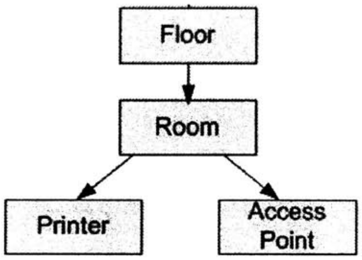


Figure 5.2 Hierarchy of Spaces and Objects within Them

---

[47] H. Samet, "The Quadtree and Related Hierarchical Data Structures," *Association for Computing Machinery Computing Surveys*, 16 (1984).

[48] Brumitt, B., Meyers, B., Krumm, J., Kern, A., and Shafer, S. *EasyLiving: Technologies for intelligent environments*. In Proceedings of Second International Symposium on Handheld and Ubiquitous Computing, HUC 2000, pages 12-29, Bristol, UK, September 2000. Springer Verlag.

[49] Leonhardt, U., Magee J. "Multi-Sensor Location Tracking", *Proceedings 4th ACM/IEEE Int'l Conf. Mobile Comp*, pp 203-214, October 1998

[50] MIT INS: http://wind.lcs.mit.edu/

[51] Estrin diffusion: http://www.cens.ucla.edu/Estrin/index.shtml

The parent-child link in the graph implies super/sub space relationships between two spaces. These divisions could be used as location entity providing a spatial reference for users (or other objects), which will themselves be defined by some supposedly well-known characterization rather than their physical properties. It is up to the location service designer to decide how to decompose the physical environment. The location service needs to maintain a hierarchical style data structure for the space tree and handles queries of spatial relationship (i.e., containment) based on this data structure.

A fundamental principle behind any location model is that each significant room doorway in a building is uniquely named. It would be unusual, for instance, to have two Room 101's in a single building. Therefore, the label of the doorway combined with the name of the building provides a useful key value (unique id) in a database table as well as an intuitive indicator of location for users.

With such a constructed space tree it is easy to tell whether the containment relationship exists between two physical spaces. The model is also capable of answering connectedness queries (i.e., "near," "next to") that exist between two physical spaces. Simple queries include, "Where am I," "Who/what is there?" and neighborhood discovery. The challenge is to extract information intuitively from the room number. Users know that room 201 might very well be located on the second floor. A simple "TYPE" can be added to a building record in a database and all associated room numbers can be topologically arranged (see the section on topological models, below).

A symbolic map, sometimes referred to as a spatial model graph (example is shown in **Figure 5.3**) or a spatial tree, represents these different levels of spaces by nodes. Information is considered to be affiliated with a location, hence linked to a node in the spatial model. Edges (or lines) between the nodes define how these places are connected.

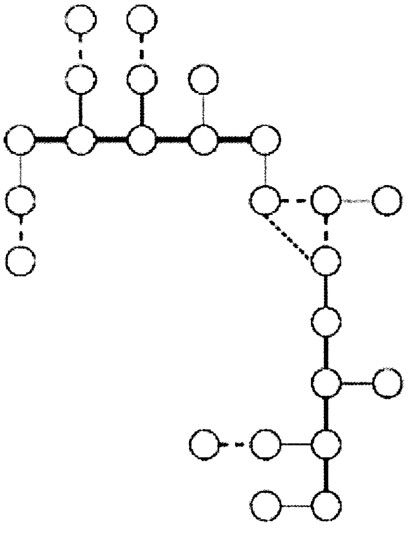


Figure 5.3: Spatial Model Graph Used

For example, a shopping mall can be represented as a set of spatial model graphs, each graph representing a floor. Each node is a place of interest (i.e., store, restaurant, ATM), which can be expanded into a more detailed scale or hierarchy, as explained above. Moreover, these nodes can have symbolic addresses or location IDs. The edges (arcs) between the nodes, in this case the stores, represent connections, which in this case are the shopping mall's hallways. Each store represented as a node may have associated secondary nodes representing shelves or products. Links between floors can be represented by special arcs (i.e., represented as dotted lines), which have specific properties indicating the type of connection (stair, elevator, escalator, ramp) and the floors they link to. In the parking lot floors, each node could represent one parking spot and the edges (arcs) are the paths to elevators and stairs. This model can also represent access privileges. For example, thin dotted lines between rooms and doors (both represented as nodes) can represent that the door is open.

Using these properties a LBS service is able to generate paths according to specific user needs. If, for example, a handicapped person wants to visit the center using a wheelchair, the navigation service will only use floor-linking arcs with an elevator or ramp connection, thus providing the user with a personalized path.

In terms of mapping out the positioning infrastructure, specifically the location of Wi-Fi APs, **Figure 5.1** shows that they are represented by small dots, but could as well be represented by a node (nodes can vary with respect to size and color, which would associate them to specific physical properties). These Wi-Fi APs dots or nodes are physical objects in space, but unlike the geometric model, they are represented in terms of relative positions, such that AP1 is "next to" or "adjacent to" Store ABC (as opposed to absolute positions of xy coordinates in the geometric model).

Liao et. al. (2003)[52] shows how to compute the graph-like structure of rooms and hallways from maps to improve the performance of location estimation and enable path prediction. In addition, being more than just a hierarchical arrangement, a spatial model graph permits intuitive traversal among node relationships, while still allowing hierarchies to be modeled.

In addition, containment relationships between 2D geometric shapes are a good way of formalizing vague spatial relationships. Simpler abstractions fail to capture complexities in the environment which are obvious to the user, while more sophisticated ones risk being too complex for the user to understand. It turns out that people are very well-suited to reasoning about and remembering 2D geometric shapes[53].

The properties of users could also be represented symbolically associated with profiling a particular user (e.g. authorizations, security constraints, etc.).

With respect to some disadvantages / limitations of symbolic location models, the main one is their inherent lack of geometric attributes and precision. A symbolic model is unable to compute

---

[52] Liao, L., Fox, D., Hightower, J., Kautz, H., and Schulz, D. Voronoi Tracking: Location Estimation using Sparse and Noisy Sensor Data. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, IEEE/RSJ, 2003.

[53] The SPIRIT project: http://www.uk.research.att.com/spirit/

distance accurately, and represent location precisely. For example, it may be difficult for the system to calculate the remaining distance between "the user" and "the nearest printer" if it is operating with them just as symbolic names. One way to do compensate for the lack of geometrical information is defining some hierarchy of information in the environment (Kuipers and Levitt, 1988; Poncela et al., 2002[54]). Another way is to divide a grid map into distinctive parts and defining their topological relation (Fabrizi and Sa.otti, 2002) [55]. Topological maps are explored in the next section.

In short, the advantages and disadvantages of mapping out the indoor world using symbolic models are:

Advantages:

- implicit representation of spatial relationships (containment, closeness). Ex: Room 200 implies a specific place in a building (second floor), intuitively distinct from Room 100 (first floor).
- supports algorithms for handling some location queries.

Disadvantage:

- lack of position precision
- sometimes inefficient to compute distance

5.3.2 Topological Maps and Location Data Models

Spatial model graphs may be seen as enrichments of topological model, modeling not only location entities as subsets or neighborhoods of space but also their structural relationships. This is implicitly the kind of model underlying the cell pavings used in cellular networks, where adjacency relationships between cells are used for the communication (and positioning) handover (**Chapter 4**) of a locatable entity from one cell to another. Adjacency is but one particular case of a spatial relationship. A complementary hierarchical model loosely underlies most of the semantic models used in directories, but is also an implicit model for the space within a building, as decomposed in floors, rooms, cabinets, etc.

Research on mobile robot navigation has produced two major paradigms for mapping indoor environments: gridbased and topological. Schroter et. al. (2002)[56] and Thrun et. al. (1996)[57] explored the problem of automatically dissecting grid-based maps through robot exploration to learn detailed features and topological layout. Kuipers and Levitt (1988) were one of the first using the concept of topological maps[58]. They defined a cognitive map on several levels of

---

[54] Poncela, A., Perez, E. J.,Ban dera, A.,U rdiales,C . and Sandoval, F. (2002). Efficient integration of metric and topological maps for directed exploration of unknown environments, Robotics and Autonomous Systems 41(1): 21-29.

[55] Fabrizi, E. and Sa.otti, A. (2002). Augmenting topology-based maps with geometric information, Robotics and Autonomous Systems 40(2-3): 91-97.

[56] Schroter, D., Beetz, M., and Gutmann, J-S. RG Mapping: Learning Compact and Structured 2D Line Maps of Indoor Environments. *IEEE Inttemational Workshop on Robot and Human Interactive Communication,* September, 2002.

[57] Thrun, S. and Bücken, A. Integrating Grid-Based and Topological Maps for Mobile Robot Navigation. Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI), Portland, Oregon. 1996.

[58] Kuipers,B. J. and Levitt, T. S. (1988). Navigation and mapping in large-scale space, *AI Magazine* 9(2): 25-13.

abstraction, where one of these levels was topological. They further extended their approach to learning a spatial semantic hierarchy of an area (Kuipers et al., 1993)[59]. Horswill (1998) constructed a system using a topological map specifically for office environments assuming angles of 90 degrees between all corridor parts[60]. Corridors can also be divided into large cells, where each cell defines a topological unit as done for the indoor navigation systems Dervish (Nourbakhsh et al., 1995[61]; Nourbakhsh, 1998[62]) and Xavier (Koenig and Simmons, 1998[63]), for example.

Topological maps are very compact representations and are usually easy to construct due to their low complexity. Another advantage of these maps is that they only contain information which hardly changes over time (rooms or corridors). Hence, they are still valid after, for example, refurnishing an office space.

Also, grid-based methods produce accurate metric maps, their complexity often prohibits efficient path planning and problem solving in large scale indoor environments. Topological maps, on the other hand, can be used much more efficiently, yet accurate and consistent topological maps are often difficult to learn and maintain in large scale environments, particularly if momentary sensor data is highly ambiguous. Overall, all these efforts must continue with increasing emphasis on wide-area deployment and larger numbers of users.

5.3.3 Geometric Maps and Location Data Models

The following is a discussion of mapping out or modeling the indoor world using geometric location modeling. Geometric location maps are made up of geometric features (i.e., points (i.e., water fountain), lines (i.e., doors, hallways), polygons (i.e, rooms, buildings)) that have metric (and non-metric) attributes.

The large and small dots in **Figure 5.1** are geometric points representing physical objects in space (i.e., RF beckons/emitters such as Wi-Fi APs) with an absolute position (xy coordinates). If the emitters represent the location of a store, they can be presented together with the corresponding WAP address or SMS number. The lines in the middle of the hallways don't represent physical objects in space, but resemble the metric characteristics of a physical object/entity (i.e., hallways) like distance.

---

[59] Kuipers,B. J., Froom,R ., Lee,W .-Y. and Pierce, D. (1993). The semantic hierarchy in robot learning, in J. Connell and S. Mahadevan (eds), Robot Learning, Kluwer Academic Publishers, Boston, MA, pp. 141-170.

[60] Horswill,I. (1998). The Polly system, in D. Kortenkamp, R. P. Bonasso and R. Murphy (eds), Arti.cial Intelligence and Mobile Robots: Case studies of successful robot systems, MIT Press, Cambridge, M A, chapter 5, pp. 125-139.

[61] Nourbakhsh, I., Powers, R. and Birchfield, S . (1995). DERVISH: An office navigating robot, AI Magazine 16(2): 53-10.

[62] Nourbakhsh, I . (1998). Dervish: An once-navigating robot, in D. Kortenkamp, R. P. Bonasso and R. Murphy (eds), Artificial Intelligence and Mobile Robots: Case studies of successful robot systems, MIT Press, Cambridge, M A, chapter 3, p p. 73-10.

[63] Koenig, S . and Simmons, R. G. (1998). Xavier: A robot navigation architecture based on partially observable markov decision process models, in D. Kortenkamp, R. P. Bonasso and R. Murphy (eds), Artificial Intelligence and Mobile Robots: Case studies of successful robot systems, M IT Press, Cambridge, MA, chapter 4, pp. 91-122.

In order to compute spatial relations (i.e., containment and intersection) geometric attributes must be present in a well-defined spatial reference system [64], such as shapes, extensions, point coordinates, etc. Geometric models define an n-dimensional space, and the locations are points in this space that can be uniquely specified uniquely and accurately represented by a tuple of numbers (x, y, z). However, there are sometimes mismatches in the meaningful precision of the coordinates in various locations. An example of a spatial (coordinate) reference system is the one utilized by LPS system MIT Cricket (**Chapter 4**).

Various coordinate systems may be used, but they must have well-defined transformations between them. For example, each floor of a building typically acts as a separate spatial reference space – two points on different floors may have the same coordinates on their respective floors, but have an unknown relationship in the real 3D world. One way to solve this problem is to allow each space to have its own local coordinate system by specifying the origin point and three axes of “x”, “y”, and “z.” This is done within the CMU Aura Project’s hybrid location model (see the section on hybrid models, below). This approach also works when buildings are more complex, for example if they have wings or towers, or if two buildings hare a parking lot or structure.

While coordinates are easy for computers to manipulate and for humans to manipulate graphically, they may not convey intrinsic meaning to humans, and listing them in text may be rather tedious. Within a geometric model, the relevant objects have to be identified, which requires symbolic information that is provided in form of attributes. This information is used to restrict a query to return only relevant objects such as rooms or buildings.

The complexity of a feature map can vary from basic models containing only walls to a complete CAD / 3D GIS model of the environment.

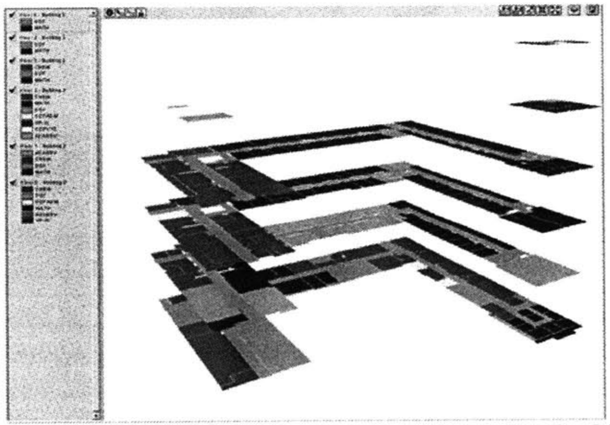


Figure 5.4: 2-D GIS (Source: MIT Facilities, Mike Parkin)

---

[64] Note, there is no such thing as a single absolute coordinate reference system ; for the sake of our argument, it could be either geocentric cartesian coordinates, polar geographic coordinates (latitude, longitude, elevation) or planar projection coordinates, as defined under a "well-known" geodetic system.

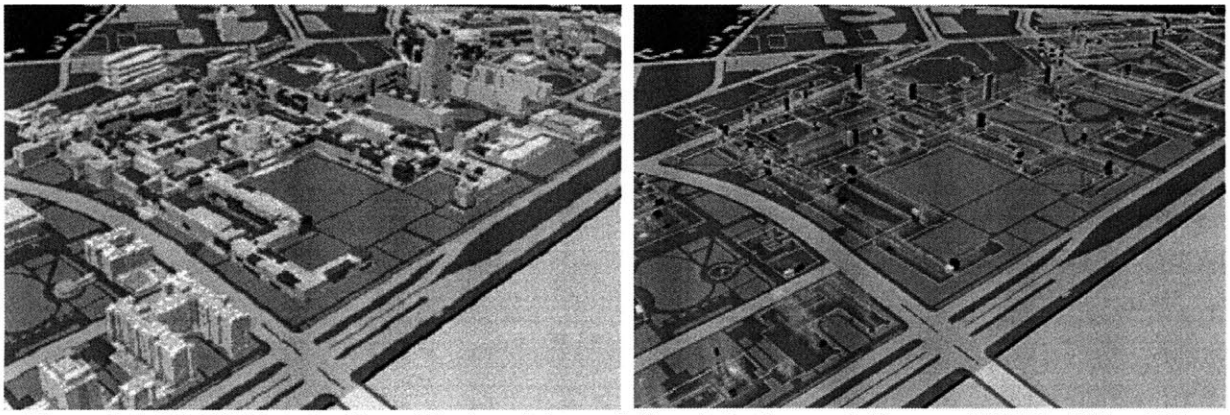


Figure 5.5: 2D GIS Data by Room Use (on right: male/female bathrooms) (Source: MIT Facilities, Mike Parkin)

In short, the advantages and disadvantages of mapping out the indoor world using geometric location models are:

Advantages:

- precise location and distance computation (due to built-in geometric attributes). Computations are typically Euclidean, and many rich relations can be computed without pre-storing them.
- indoor location positioning systems that rely on the geometric model generate streams of position fixes that are independent in the sense of describing a location without external reference. There is a shared reference grid for all located objects, which can be transformed into a relative location, if needed.
- allows to define points or areas for which there is no name in the hierarchical name system

Disadvantages:

- hides hierarchical relationships (so it needs extra specification to enable deduction of spatial relationships).

5.4 Navigation and the Indoor World

A navigation LBS service needs some kind of knowledge or representation about its environment. These representations are usually in the form of maps, which have geometrical and topological properties, capturing the properties of the environment. Furthermore, they can also be symbolic, in the form of labeled entities, which are typically used for task specification. This knowledge is often referred to as symbolic reasoning allowing decision making about the world. Symbols with geometrical properties redirect the immediate surrounding of the object or person being navigated in order to allow safe navigation. In indoor navigation, these types of symbols include walls, doorways, or obstacles such as tables, chairs, and people. On the other hand, a model of the large-scale structure of the area is required to enable planning of routes to fulfill an entire path planning.

Overall, indoor navigation is going to be more often the desire than the need until an infrastructure of positioning/navigating aids (i.e., anchors) is in place.

5.4.1 Navigation using Symbolic Maps

The following is a discussion of how relative positioning (**Chapter 4**) can be used for navigation using symbolic location modeling. Direction sensing based on a symbolic model is based on the 'trigger effect' such that when user 1 left area AP1 and entered area AP2 (few seconds later), the direction can be deduced that user 1 is moving up the corridor (passing AP2 going towards AP3). In terms of giving navigational instructions to the user, they can be given in terms of 'symbolic reasoning' (as opposed to 'geometric reasoning' requiring metrics), for example, "*when you pass X (the ABC store), you will see Y (the ATM) on your left.*"

5.4.2 Navigation using Topological Maps

Topological maps are suitable for indoor navigation not only for mobile users but also automatic/robotic wheelchairs (or Segways), which require higher precision and accuracy. The main structure of the map containing qualitative information about the large scale connectivity of the environment is redirected using a spatial model graph (**Figure 5.3**). Nodes stand for important places in the environment and locations where a change in the navigational strategy occurs (these nodes are represented in the middle of the hallways in **Figure 5.1**). Hence, there has to be one in front of each door, at each corridor crossing and at other places of interest. Each node has a location in a fixed coordinate system. The edges that connect these nodes can be of three different types: room, corridor, door.

To enable navigation, nodes in corridors are in the middle of the two walls. The ones in front of doors are aligned with the center between the door posts. Further, nodes in rooms are positioned at places that are important for the navigational task. This placing allows the navigation system to effectively keep track of its position and orientation. Nevertheless, these coordinates need not to be very accurate, because the nodes in combination with the robot position estimate are only needed for task switching. Context information and a predefined world model enable the coordination scheme to switch between subtasks. These representations constitute symbols, on the basis of which the system makes decisions. These symbols must be anchored in the real world, requiring the capability of relating to sensory data from the LPS systems discussed in **Chapter 4.**

Note that actual guiding of the robot through a door, for example, is controlled by the behaviors which extract the precise location of the door posts from sensory data. Local geometrical representations parameterize these behaviors.

Topological maps are harder to use for navigation purposes than metric maps, because only limited knowledge about the object's surrounding is available. Nevertheless, it also contains some minimal geometrical information redirected in the properties of the different nodes, which defines their location in the world.

5.4.3 Navigation using Geometric Maps

The following is a discussion of how absolute positioning (**Chapter 4**) can be used for navigation using geometric location modeling. This type of precise navigation is essential for wheelchairs or a Segway (refer back to **Chapter 2** for the Product Finder and navigation use cases).

In **Figure 5.1**, the large dots are placed at intersections in the hallways representing the possible change of directions (right, left, straight). The small dots are placed at both sides of doorway's entrance (doorframe) in order to achieve precise entry opening essential to, for example, an automatic wheelchair, so that it doesn't collide with the doorframe.

Navigation is achieved by calculating the distance traveled with respect to the distance remaining in reaching the destination. The ISO TC/211 standard for navigation, "19133 Geographic information -Location based services tracking and navigation[65]," describes the data types, and operations associated to those types, for the implementation of tracking and navigation services. This international standard is designed to specify Web services that may be made available to wireless devices through Web-resident proxy applications, but is not restricted to that environment. The OpenLS APIs (**Chapter 6**) seems focused on providing the underpinning for navigation (and network analyses). In this context, I see the DCT API providing a dynamic WayPointList, and DCT attributes being able to serve the roles of AOIList, LocationList, and similar point-derived features.

Also, since located objects become mobile, the need to identify localized objects (to describe their movements and to estimate their future positions) is also of significance. For this purpose the elements of probability theory[66], statistics and machine learning[67] are used.

5.4.4 Hybrid Location Models

Mapping infrastructure providers (i.e., content providers) have to figure out how to seamlessly provide location content utilizing various types of local location model. A mapping infrastructure combining both the geometric and symbolic location models makes it possible to support LBS services at all relevant scales, in all types of positioning type environments. Proposed approaches build upon hybrid space models, federating various interpretations of location and location-sensing technologies[68].

In order to get the interconnections between places, it is necessary to have relations between the objects involved (**Figure 5.5**). Some relations are modeled implicitly when the geometry is modeled, e.g. which rooms are next to each other (topology). However, this may not be sufficient, if the geometric model lacks information about doors between rooms. In this case, this relation has to be explicitly modeled. Since all this information can be modeled in geometric location model, the symbolic location model (spatial model graph) needs can be downloaded at

---

[65] ISO TC/211 www.isotc211.org

[66] Schiele, B., Pentland, A.: Probabilistic Object Recognition and Localization. In ICCV'99 International Conference on Computer Vision, Greece, September 1999.

[67] Addlesee, M.D., Jones, A., Livesey, F., Samaria, F.: The ORL Active Floor. IEEE Personal Communications, October 1997, Vol. 4 No. 5.

[68] Coulouris, G., Mitchell, S., Naguib, H. "Middleware Support for Context-Aware Multimedia Applications", Proceedings of the third International Working Conference on Distributed Applications and Interoperable Systems, IFIP Conference Proceedings, vol 198, pp 9-22, September 2001

special locations with connection to the infrastructure and from then on, it can be used in its ad-hoc network environment.

The integration of the geometric and symbolic location models is achieved through bundling nodes in the space tree with geometric attributes. The resulting tree is called geometric space tree, which has geometric attributes embedded into the nodes that include:

- Shape (indicates the geometric shape of space – cylinder, cube, sphere, etc)
- Extension (for specifying volume along with the shape attribute)
- Origin
- Rotation Matrix

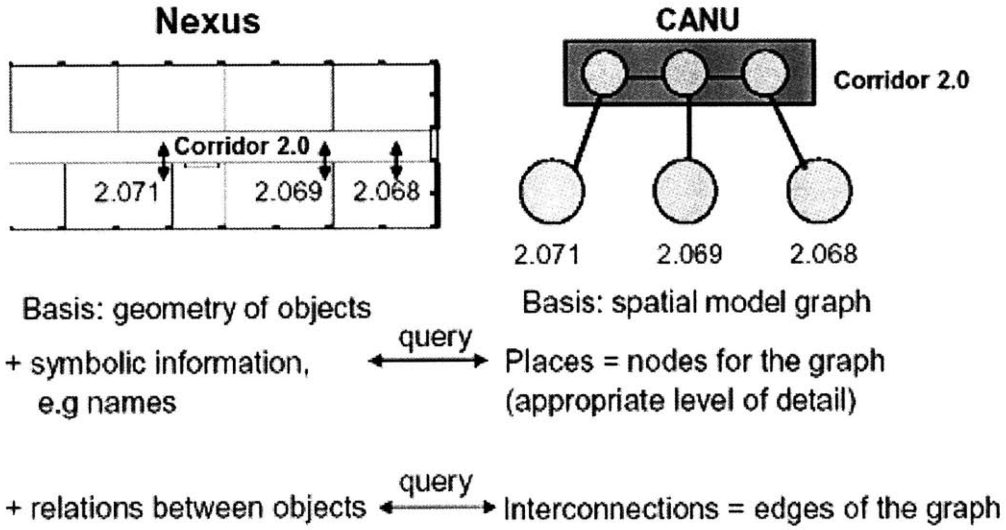


Figure 5.5: Integration of NEXUS (geometric) and CANU (symbolic/topological) Location Models (Image Source: IPVR, University of Stuttgart)

Another possible solution to provide seamless mapping (location) content for LBS services is the layered model architecture of location proposed by France Télécom[69]. Fulfilling the requirements of combining several location models types, the researchers have created an architecture template (shown in **Figure 5.6**, where *locus* is user or object and *loci* is a locatable entity such as a point of interest) that draws a conceptual analogy to the layered abstraction levels of network protocols (ISO network layer stack). Based upon such a general framework, location should become integral to all software (service) infrastructures (i.e., service/device discovery and management), as a basic attribute of all networked entities, supporting location-based queries (**Chapter 6**).

---

[69] Flury,T., and Privat, G. An infrastructure template for scalable location-based services. France Télécom R&D, DIH/OCF 2003.

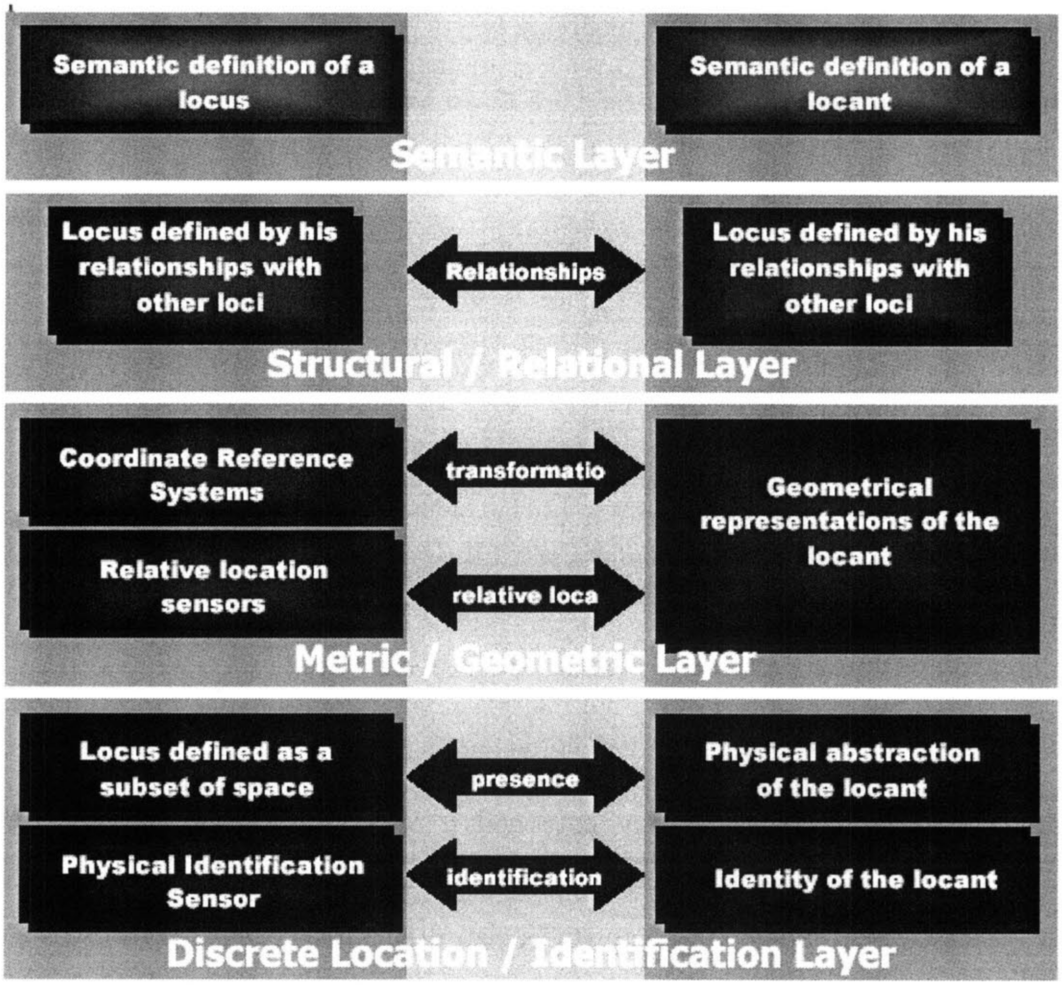


Figure 5.6 Location Layers Architecture Model (source: France Télécom)

In this location layer model, the bottom layers are closest to the physical properties of space, and as the higher layers get more abstracted away and closer to concepts understandable by users. This is similar to the ISO network layer model, moving up from physical connection, to MAC addressing, to IP, then to DNS and possibly UDDI addressing, in network-based identification protocols. This location layer model consists of two vertical domains of representation, orthogonal to layers. The flow of information is vertical through each of these with using horizontal relations at each layer.

By a loose analogy with the physical layer of networking protocols, these layers correspond to the physical location-sensing and identification technologies used, which are much more varied than physical networking technologies. They appear in **Figure 5.6** as the lowermost boxes of both the discrete location and metric layers. As such, they comprise technologies that identify users (*locants*), and technologies that locate them in space, which may be the same, but not necessarily. It is only by combining the two that tracking a user is made possible. A sonar/radar-like or vision-based ranging system can be used as a relative positioning technology, but may have to be used with an secondary identification technology for actual tracking.

The next section addresses data location modeling languages that are used to interact (i.e., query) the world model presented above. Note that these modeling languages are mostly applicable to the geometric location model, but since they are based on XML, hence, are extensible, they are adaptable for the symbolic location model, too. Overall, these languages are data APIs that are used as the communication exchange protocol via the Web.

5.5 Location Modeling Languages

With respect to mapping location using a geometric data location model, there are a number of shapes used to represent a geographic area that describes where a mobile user is located. There are additional shapes that are required for advanced LBS services.

The standards bodies for geographic data for advanced LBS services such as routing, geocoding, coordinate conversion, and map display are the LIF, the OGC and the ISO TC211 working group. The current public XML specification defining geography from these groups is GML[70].

Overall, the requirements for a modeling location language are:

- Decoupling the application from the internal representation of the world model (model the world independent of a particular application).
- Describe the geometry of objects relative to different coordinate systems.
- Express symbolic information, i.e. names and descriptions of objects.
- Express certain relations (i.e., *inside*, *overlaps*, *includes*, *excludes* and *closest*) between objects.
- Support the description of objects in different levels of detail/scale

Decoupling the application through a query language from the internal model representation allows tohide the different levels of detail of models as well as the necessary model representations. The model can be queried, using a querying language (such as OpenGIS Filter Encoding specification (**Chapter 6**) that provides a useful abstraction. OpenGIS GML, discussed next, can be used to describe geographic features' properties, which includes geometry. It can also be used to model symbolic information (features' attributes) and to encode spatial relations that can be either symbolic or geometric.

5.5.1 OpenGIS Geography Modeling Language (GML)

GML is an XML encoding for the transport and storage of GI, including both the geometry and properties of geographic features. Furthermore, GML is an XML representation of OGC Simple Features, which is a specification for vector based map-content (geographic features) for GIS systems.

GML is build on a hierarchical vector model, with feature collections compromising features that can comprise more features. The GML geometry schemas support points, lines and polygons as

---

[70] OpenGIS GML: http://www.opengis.org/techno/specs/00-029/GML.html

its basic geometric elements, but also geometric collections such as multipoints, multilines and multipolygons.

Regarding modeling mobile objects, GML has a schema called dynamicFeature.xsd which provides support for time dependent features. This schema includes a class called MovingObjectStatus which is effectively the State of a moving rigid body (location, speed, direction). This was inserted to support LBS and other requirements.

GML can be expanded via application schemas to add a “PointCircle,” “PointEllipse” and “PointArc” elements to accommodate the LIF “CIRCLE,” “ELLIPSE” and “ARC” elements, which are used to describe error estimates of mobile device location. These additional geometries added for LIF are additional geometries and could be used for any purpose. There is also a GML component (in the GML namespace) called observation which can be used to denote a measurement - this could also be used in conjunction with a geometry to define an observation respecting an objects location.

GML can also be used to model hierarchical/symbolic objects (based on a non-geometric location model). By hierarchical non-geometric location model, it is assumed that an object is located relative to another object. This can always be done using GML properties which can be associations.

```
<abc:Building gml:id = "building1">
<gml:name>Store ABC </gml:name>
<abc:frontsOn xlink:href = "#r1"/>
<abc:numStories>3</abc:numStories>
</abc:Building>
```

Other kinds of relationships including hierarchical containment can be constructed. In this case a dictionary XML Schema (dictionary.xsd) of property elements (e.g. <abc:frontsOn>) should be created that express the desired feature relationships. This schema is then used in the application schemas to construct the particular relationships that are desire.

GML is positioned as an open data exchange standard, well suited for transmitting small to medium-sized volumes of information. GML is usable with all standard XML tools. Of particular note in this respect are the tools designed to filter XML (XSL) and to turn XML into a visual presentation (XSLT). Using the XSL tools, a fully functional GML database can be published into more limited versions. For example, in order to satisfy regulatory requirements, a subset of the data, perhaps with lower fidelity, can be automatically extracted. To share data with a supplier who is also a potential competitor, the data can first be filtered and adjusted on the basis of what the supplier needs to know.

5.5.2 NEXUS Augmented World Modeling Language (AWQL)

The Nexus augmented world model is described using the Augmented World Modeling Language (AWML) [71]. Location-aware applications may query the current state of this model by using the Augmented World Querying Language (AWQL) and, as a response, receive information about the model described by the AWML. Both languages are defined using XML schemas. The response to queries are serialized in XML, specifically AWML. As XML becomes more popular as format for data exchange, and many newly developed data formats are XML based, using an XML data format for location-based services has the advantage that such data can easily be embedded in AWQL or AWML.

Objects in AWML have attributes that give their geometry relative to some coordinate system. GML is used for geometry description of common data formats (i.e., '*GML: polygon*'). The objects belong to classes that are structured in a hierarchical class schema i.e. a church is a building, which in turn is a static object and a Nexus object. NEXUS uses several different coordinate systems, e.g. WGS84 coordinates (used by the GPS system), Gauss-Krüger, and UTM coordinates. Note that GML does not define a fixed coordinate system to be used. Instead, a spatial reference system (SRS) has to be given, relative to which the coordinates are defined.

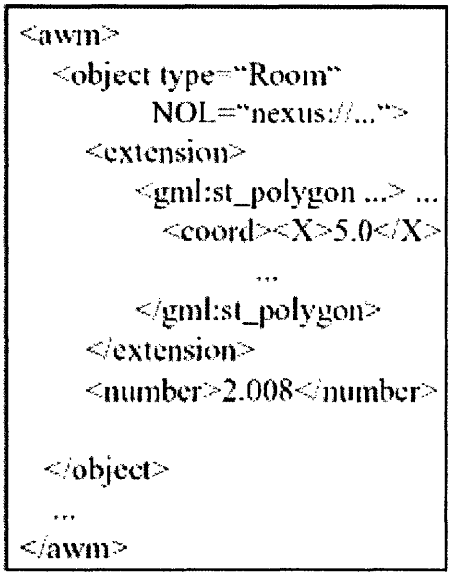
In addition, AWML not only models geographic location and the geometry of objects, but also symbolic descriptors of the objects such as room numbers and explicit relationships between objects, such as the *part-of* relation. This is especially important when linking together different parts of the model that may be supplied from different providers.

In summary, NEXUS AWML's features are:

- Object geometry (GML)
- Coordination system (absolute systems: WGS84, UTM; relative systems)
- Symbolic description (i.e., IDs, names, room numbers)
- Relationship between objects ('part-of relation')

---

[71] Nicklas, D., Mitschang, B. The NEXUS Augmented World Model: An extensible approach for mobile, spatially aware applications, 7th International Conference on Object-Oriented Information Systems, 2001



```
<awm>
  <object type="Room"
          NOL="nexus://...">
    <extension>
       <gml:st_polygon ...> ...
         <coord><X>5.0</X>
         ...
       </gml:st_polygon>
    </extension>
    <number>2.008</number>

  </object>
  ...
</awm>
```

Figure 5.7: NEXUS Augmented World Modeling Language (AWML)

5.5.3 Navigation Modeling Languages

LIF MLP API

LIF MLP API has elements that enable the positioning system to return values for direction and velocity. The DIRECTION element expresses position in degrees (with north as 0). The VELOCITY element expresses the speed of the mobile device in meters per second. Both are only present if the positioning method used can be used to calculate it (positioning using a cell ID does not make it possible to calculate the velocity, for example).

Navigation Markup Language (NvML)

NvML sefines points, routes, information elements, and child elements under those that can hold information of various kinds. The points are intended to be along a route, which can be defined by an external entity. The route could also be pre-filtered depending on the user's interest. The concept of routes is as difficult to define as that of areas. In the context of NvML, it implies a vector with duration in time.

5.6 Conclusions

Most LPS systems (**Chapter 4**) allow independent determination of position, yet this information must be communicated by separate means in order to be shared. It is difficult to exchange information between these systems as they are today (being self-contained and vertically integrated), as it would require data conversion, among other things. Nor is it possible to notify the applications on one system about information changes based on the information sensed by another system.

Overall, a common location model is essential for indoor LBS application integration and service bundling (**Chapter 6**), but it should not necessarily reduce modularity. In order to make the exchange of information more interoperable, we addressed open standards from LIF/OMA and OGC, which are key to making the mapping infrastructure seamless across any system and application.

In order for a location model to be practical, a federation should exist, with more application-oriented models dependent on more fundamental models (but not vice versa). The development of detailed models is a costly task (especially when extending them to detailed world models). Therefore, different applications should be able to share the same model information. Having such a common model may increase interoperability between applications and make new classes of applications possible (due to easier integration). The basic requirement for such an approach is a common language for describing and querying location information. An example of such a query language is Nexus's AWQL and OpenGIS Filter Encoding specification (**Chapter 6**).

Such means as the mapping infrastructure combine several complementary models of physical space (topological, metric, Euclidean, symbolic/semantic) and make it possible to support LBS services at all relevant scales, in all kinds of environments. A template for this infrastructure may draw a conceptual analogy to the layered abstraction levels of network protocols (the ISO model). Based upon such a general framework, location should become integral to the software (services) infrastructure (i.e., all the service/device discovery and management components) discussed next in **Chapter 6**, as a basic attribute for the communication (querying) among all components.

Despite the developments presented in this chapter in terms of location models and modeling languages, challenges still exist regarding the modeling of indoor location and navigation. These challenges include:

- *Managing complexity and scalability*: As models increase in complexity, the management and integrity of the information becomes a critical design issue. In addition, the design of a model should not only take into account the potentially large number of entities in a single environment, but also factor for multiple environments linked together.
- *Transient environments and aggregation of sensor data*: Designing a model that successfully bridges the difference between administrative, social and home environments is challenging. Focusing the design on a single environment may obscure difficulties when applying it to another environment type. Many environments will support one or more differing LPS systems. Aggregation of this multiple sensor data would rely on an abstract location model not directly connected to or dependent upon a particular LPS.
- *Inference beyond position*: Whilst determination of position remains important there is potential for greater contextual inferences to be made from a model in terms of representing conceptual, logical and physical connectivity.
- *Ontology for location*: The decision of how to describe space is not a trivial matter, however, a common means to represent location across various different models may be useful. Semantic location information can be powerful for many tasks, but it remains an open

problem to gather and represent both semantic position tags and detailed geometric location in a single system.

- *Open and extensible model*: The task of providing location information for the model should not rely solely on a single source. The ability for other providers to supply additional information is desirable. In order for a model to evolve along with changes in the environment it and the sensing technologies employed it must be easily extensible and adaptive.

It is important to realize that symbolic need not be less accurate/preferred than geometric as the real issue is whether the underlying data model and services is based on symbolic (adjacency/topology) or geometry (see **Chapter 6** on LBS services). For example, using location to snap to a room or floor and then give services (what's close or navigation) based on the being in the room versus using geometry and Euclidian distance as the basis for judging what is close/far (without caring whether there are walls/buildings in between).

In most cases where a simpler approach (i.e., a symbolic location model of a diagrammatic floorplan layout) might suffice and be significantly cheaper for some basic LBS application, there are additional LBS applications of interest that can only be implemented with a geometric location model, especially those that require arithmetic calculations (i.e., distance measurements). Hence, it is important to consider whether the simpler approach will suffice for all applications that may be needed in the long term. If not, it makes sense to implement a mapping infrastructure (location model) that will handle all applications from the simple to the sophisticated. And, depending on what kind of LBS application is to be provided, different semantics need to be considered, as well.

Overall, the mapping infrastructure must be extensible enough to federate different ways of modeling and abstracting away physical space[72]. It should be sufficiently accurate to fulfill the needs of all sorts of specific LBS applications (**Chapter 2**) while being general enough to be independent of LPS systems and sensor technologies (**Chapter 4**). There are two goals that should be reached to fulfill these requirements. The first one is to handle and generate location queries (which, in addition to a position filter may also include events[73]) forwarded to the interested users. These events may also be filtered by an intermediate service to retain only the relevant activities (i.e., moves), depending on the location model and the application concerned. The OGC Filter Encoding specification (**Chapter 6**) is an example for a standard way of querying and filtering using XML. The other is to respond to location queries based on location events (i.e., location-based triggers – Buddy Finder and Product Finder). The response has to be provided in real-time, preferably in a standard way for other services to integrate when/if needed.

---

[72] Leonhardt, U., Magee, J., Dias, P. "Location service in mobile computing environments", Computers and Graphics, vol 20 n° 5, pp 627-632, 1996

[73] Spiteri, M.D., Bates, J., "An architecture to support storage and retrieval of events", Proceedings of Middleware 98, IFIP International Conference on Open Distributed Processing, Lancaster, UK, September 1998.

CHAPTER 6

THE SOFTWARE (SERVICES) INFRASTRUCTURE

6.1 Introduction

As mentioned in **Chapter 1**, the role of the infrastructure is crucial when deploying LBS applications. The software (service) infrastructure is the fourth essential type of the overall infrastructure, and might be the most important since it's tied to the user directly. More specifically, this infrastructure determines what the user sees on his mobile device and how he/she interacts with it.

As reviewed in **Chapter 2**, most current applications are based on proprietary technologies, closed-architectures, and 'full-systems.' While a good start, these standalone ("stovepipe[74]") applications are unlikely to make a large impact on the marketplace for reasons dealing heterogeneity stemming from lack of interoperability, which prevents application integrations and service-chaining. This means that there is no way for an application to share, access, or control the sensing resources without knowing the sensor and the network specifications. Moreover, there is no consensus on common standards for designing these applications, and, as a result, there is no common software platform on which to build such applications that would enable easier deployment as discussed.

This situation will predominate for some time until the understanding and language required to define open and standardized interfaces at each level (pertaining to each infrastructure type) of the system has been developed. In **Chapter 3**, we stressed the significance of the standardization process and open interoperability standards in relation to achieving the full market potential by service-chaining.

However, as with the outdoor world of LBS, the standalone application approach currently present in the indoor world will change, as existing and emerging standards enable Web proxies and servers to get location information directly from the cellular network. Moreover, technological driving forces such as Web services will enable easier integration of the various software components and result in interoperability.

Hence, in the light of these market and technological driving forces, the purpose of this chapter is to outline which components need to be standardized to enable interoperability, which will accelerate the deployment of any kind of LBS service. In addition, we explain which issues are more important, what are key choices, and tradeoffs in developing these services in terms of the software (service) infrastructure. In addition, this chapter explores what decisions impact the timing and practicality of integration and interoperability.

---

[74] "Stovepipe" is a metaphor commonly used to describe systems that are integrated "from top to bottom" but isolated laterally, i.e., from other systems. A stovepipe system might be a system from a single vendor or it might be a system built by an integrator, but it is not an open system.

Overall, the integration of the various components from each of the infrastructure types (and from different providers), highlights the complexities of effectively delivering indoor LBS services and raises a number of questions:

- How do two providers integrate their services?
- Is it possible to integrate any kind of services?
- How do these services exchange information?
- How do you initiate\request a bundled service?
- How does a bundled service adhere to industry regulations/standards? (technical viewpoint)
- How do you achieve authentication and authorization between the services?

Before getting into the discussion of the various components, it is useful to understand how LBS services can be designed. As mentioned in **Chapter 1**, LBS service can be classified into three types, which are in an increasing level of magnitude in terms of implementation:

- Simple or basic – users manually enter their location (i.e., address, phone number, place)
- Location-aware – location determined automatically (i.e., 'triggered')
- Context-aware (or ubiquitous) – adaptivity to user's activities and events

For discussion purposes, this chapter also considers the Buddy Finder and Product Finder applications. Both of these applications can be achieved using the "simple or basic," "location-aware," or "context-aware" form. It is up to the developer to decide what form is best, which will depend on what the customers want, and how much they are willing to pay for it.

It might be logical to start with the most basic form of the service and offer it to customers for free, which will, in turn, lock-in a large enough customer base. With good marketing this approach will attract new customers that would be required to pay for this service. The original customers (as well as new ones) would have the option to upgrade to the next form of the service ("location-aware") for a charge. These business scenarios are explored in detail in the next chapter, **Chapter 7**, where we take the perspective of a service designer and examine strategies to bring indoor LBS services successfully to market.

6.2 "Simple or Basic" LBS Services

A simple or basic LBS service is where the user asks the service to give him a map of a specific shopping mall without specific location determination (user knows he/she is in a particular mall for which he is requesting the map). Or, the user manually enters his/her location as an address, phone number, or place. The former doesn't agree with the definition of a LBS service[75] that was presented in **Chapter 1**, but it could be developed as a startup for the latter.

A good example of this basic or simple service is the MapQuest.com's 'Business Name or Category' search. In its most basic form, this service allows the user to enter, for example,

---

[75] "A location-based service, or LBS, is the ability to find the geographical location of a mobile device and provide services based on this location" (Kurt Buehler, OpenGIS Consortium, OpenLS Initiative: *http://www.openls.org*)

"shoes" as the 'category' and "Boston, MA" as the location-based search criteria. The result is a list of stores in Boston that sell shoes. In a more all-inclusive service, the user then has the option of getting a map of a particular store's location, as well as entering his/her address location to get specific directions to that store. Another example is the LBS services Guru from Tre (3), the Operator 3 in Sweden. Guru is a map service where the subscriber can get color maps on their 3G phones using a number of services such as: *"Here I am [i.e., in the mall] – shows a map on where the subscriber is located...," "Show a place – shows a map over a defined area," "Where is it (Find POI) – Yellow pages."*

Similarly, such a service for an indoor setting (inside the shopping mall) would use "Shopping Mall ABC" instead of "Boston, MA" as the input for location, using the same category "shoes" as the search criteria. This would return a map of the shopping mall layout (similar to the maps on stands throughout the mall), highlighting the stores that sell shoes. Actually, if a map with the appropriate legend can be designed for display on a small screen of a mobile device, there would be no need to highlight specific stores or place points/symbols on them indicating the search result.

What is significant here is that in all these case, there is no LPS system involved, hence, no real need for the positioning infrastructure. In addition, determining user's location (by manual entry) is a minor value-add to the core service (e.g. the MapQuest.com service offers directions to restaurants and other information without automatic position determination). In these cases the positioning capability may be so marginal that it could be excluded or made optional to counter privacy concerns (privacy is one of the design considerations for an LBS service, explained later in the chapter).

6.2.1 Service Use Case

For the Product Finder the service would provide information about the physical environment (the shopping mall) and spatial relationships between objects (i.e., the store with the product) and their locations. A typical service use case (**Figure 6.1**) is to have a user wanting to find the nearest store that has a product of interest. The service request/respond cycle would involve three separate actions (requests). Note that for the user, the interface can be designed so that the user only executes one action (enters his/her location), but behind the scenes, there are two additional requests that are executed.

Since this is a 'basic or simple' type of an LBS service, there is no LPS system involved (hence, no need to model user's positioning data). As a result, the first action is having the user enter his/her location to the People Location Service. Second, using the response from the first request, the user (or the service's interface) asks the Product Service for the locations and names of available products (stored as part of the mapping infrastructure's content/baseline data). Then, the service queries the Space Service for the distances between all the products and his/her location, which finally identifies the nearest store (where the product is) as the one with the smallest distance.

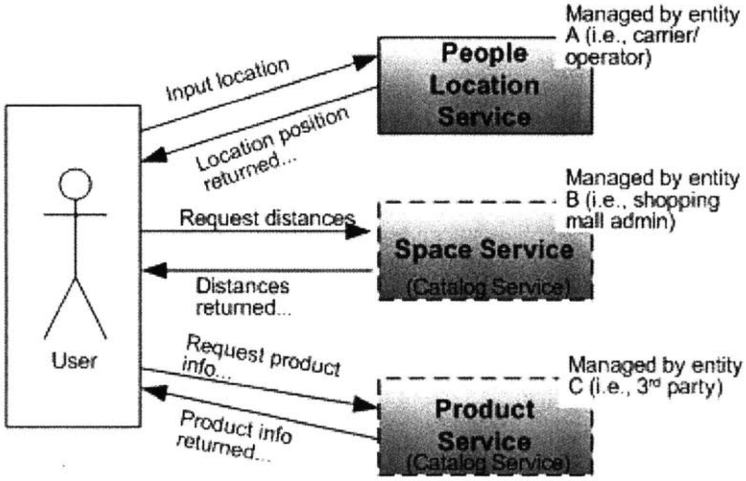


Figure 6.1 Service Use Case for a "Basic or Simple" LBS Service(s)

The requirements for implementing this "basic or simple" LBS services are as follows. Since there is no LPS system, the mapping infrastructure (**Chapter 5**), does not need to model user's position data as it would be expected with a "location-aware" service (see below). Nevertheless, instead of having a LPS system determine the user's location, a user, as described above, would manually enter his/her location, preferably in terms of relative positioning such as "I am in Shopping Mall A, Store ABC." This would require the underlying data location model to be of the symbolic type (or a geometric model that can map/reference symbolic meanings into absolute/geometric ones).

Similarly, the Product Service can return the location of a products as "*in store ABC*" or "*xy coordinate*"). In the former case, a symbolic location model would be used. In the latter case, a geometric location model is needed to give precise location. Furthermore, the absolute positioning output in the latter case would most likely be presented to the user using symbolic reasoning as it was used in the former case ("*you are in store ABC*"). The need for a geometric model, however, is especially evident with the Space Service, which needs to handle queries of spatial relations (distance) between locations of the user and the store/product by modeling the physical environment using metrics. Since there is a potential that either service request will use a geometric location model while the other a symbolic location model , there is the need for a hybrid location model, which was explored in **Chapter 5**.

Also, the mapping database would store and model the content/baseline data such as location of stores (this could resemble a CAD or GIS drawing, see **Chapter 5**). Products could be either stored with their own location (as data points) or aggregated as attribute data for the store locations.

In most cases where a simpler approach (i.e., relative positioning and symbolic location models) might suffice and be significantly cheaper for a basic service, there are additional services of interest that can only be implemented with a more precise LPS. Hence, it is important to consider whether the simpler approach will suffice for all services that may be needed in the long term. If not, it makes sense to implement a positioning infrastructure that will handle all LBS services from the simplest (based on containment/adjacency) to the more sophisticated (dealing with distances). Hence, an absolute positioning LPS system would be a wise choice as the initial (and final) positioning infrastructure investment.

6.3 "Location-Aware" LBS Services

As mentioned in **Chapter 1,** a "location-aware" type of a LBS application is considered to be more user friendly as lot of typing by the user can be avoided since the application knows his/her location and act upon a specific location automatically. Furthermore, end applications can take advantage from location information by partially automating user queries. Location-aware applications can exploit this information in a number of ways: proximate selection, location queries and commands, and location-triggered actions.

A Buddy Finder and Product Finder applications, for example, might require a personalized user profile to make the service more productive and user friendly. The content information that is used to find a friend could be, for example, the person's interests, and to find a product, the product's brand and price. BigTribe has developed a unique adaptive personalization algorithm that recommends locations (restaurants, stores, parks, etc.) you'll enjoy visiting and services (taxis, movie tickets, restaurant reservations, prescription refills, etc.) you'll use. It works with connected mobile devices, such as web browsers and cell phones, and disconnected devices, such as PDAs. Another example is the *ParcTab*, a Personal Digital Assistant (PDA), which uses an infrared-based cellular network for communication. The infrared transmissions from ParcTabs can be used to determine their locations in the same way as Active Badges are located.

These location-aware services have adaptive menu structures for applications, ones that automatically filter content and navigation options based on user's location and/or other location-related information. An example of an adaptive LBS service is when a user is automatically presented with a food menu on his cell phone when he walks into a cafeteria, without having to click and find menu options. Moreover, rather than communicating with a customer via a mobile device, a store could have computer screens distributed around the store and, as a customer walks up to one, it knows the customer and displays relevant information. (This kind of interaction with computers that know the user's location is called pervasive or sentient computing.)

Also, AT&T Wireless announced an enhancement to its mMode service. Included in the offerings is access to movie listings and ticket purchase based on the phone's location (we assume this to be a "simple" LBS service requiring the user to enter his/her location). Another offering soon to be launched is Match.com location-based dating. This service is definitely a 'location-based trigger' application that will require location-awareness of all the users using this service.

Overall, for these types of location-based trigger applications to occur, the position of the user needs to be known, hence, the need for a LPS system. Moreover, this is an active application, or a real-time service, which may need to handle thousands of spatial updates per second, a requirement factor that is explored later in the chapter.

6.3.1 Service Use Case

In contrast to previous type of a LBS service, a 'location-aware' service does include a LPS system that automatically locates the user (the user doesn't need to enter his/her location as in the previous case). The request/response cycle (**Figure 6.2**) to get the distance and location of the nearest product is as follows. First, the user asks the People Location Service for his/her current location. Second, using the response from the first request, the user asks the Device Service for the locations of the product that the user is looking for (part of the mapping infrastructure's content/baseline data). Then it queries the Space Service for the distances between all the stores that these products are found in and his/her location, which finally identifies the nearest store (with that product) as the one with the smallest distance.

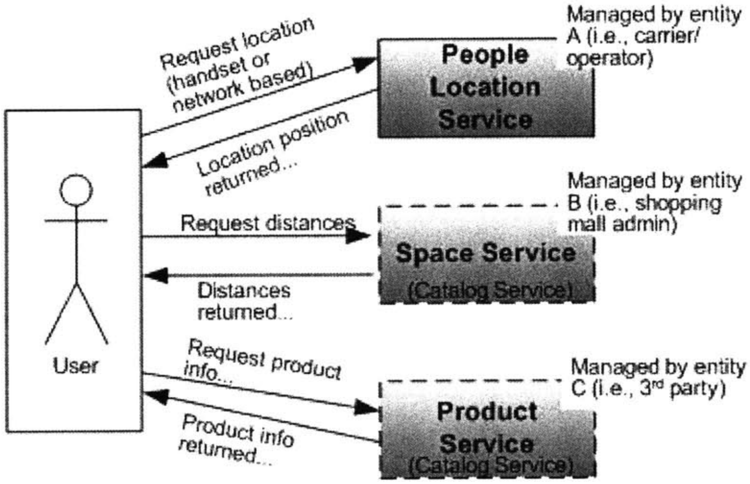


Figure 6.2 Service Use Case for a "Location-Aware" LBS Service(s)

The requirements for implementing these simple services are as follows. In terms of the positioning infrastructure (**Chapter 3**), depending on how the People Location Service is designed, it might return the location of the user using relative positioning ("*you are in Store ABC*") or absolute positioning ("*you are at xy coordinate*"). Furthermore, the absolute positioning output in the latter case would most likely be presented to the user using symbolic reasoning as it was used in the former case ("*you are in store ABC*"). Similarly, the Product Service can return the location of a printer as "*in store ABC*" or "*xy coordinate*"). In the former case, a relative LPS system (i.e., Active Badges) is effective. For the latter case, an absolute LPS system (i.e., MIT Cricket, MS RADAR) is needed to precisely locate the user.

The requirements associated with the mapping infrastructure (**Chapter 4**) are directly tied to the positioning infrastructure. Depending on which type of LPS system is used (relative or absolute) will determine the type of the data location model (symbolic or geometric) needed to model that position information. The need for a geometric model is especially evident with the Space Service, which needs to handle queries of spatial relations (distance) between locations of the user and the printers by modeling the relevant physical environment using metrics. Since there is a potential that either service will use a geometric location model while the other a symbolic location model , there is the need for a hybrid location model, which was explored in **Chapter 5**.

Note that symbolic location models need not be less preferred than geometric location models for services that require higher accuracy. The real issue is whether the service needs to answer to a request such as "what's close?" based on adjacency/topology or geometry of objects/users. For example, using location positioning to snap to a room or floor and then give this "*in the room*" location to the service based on the user being in the room versus using geometry and metric distance as the basis for determining what is close/far (without caring whether there are walls/buildings in between for navigation purposes).

In short, the first step for a service designer is to determine whether the service is "simple/basic" or "location-aware." It the latter, what needs to be determined is what kind of positioning output is necessary for a service and, hence, the LPS system type (absolute versus symbolic). This, in turn, will determine how the positioning information will be modeled (geometric versus symbolic location models). However, it important to keep in mind that even if the People Location Service doesn't need absolute positioning, hence no need for a geometric location model, other services such as the Space Service that needs metrics for distance calculations might supersede the People Location Service to use the other alternatives.

A real-world example of a location-aware type LBS service is the shown in **Figure 6.3**. Here, the user access the application server, which coordinates the handling of requests, the retrieval of information, and so on. The position is retrieved from the location positioning system and the data and presentation format comes from the database; then, they are combined according to the rules in the application and the presentation is returned to the user.

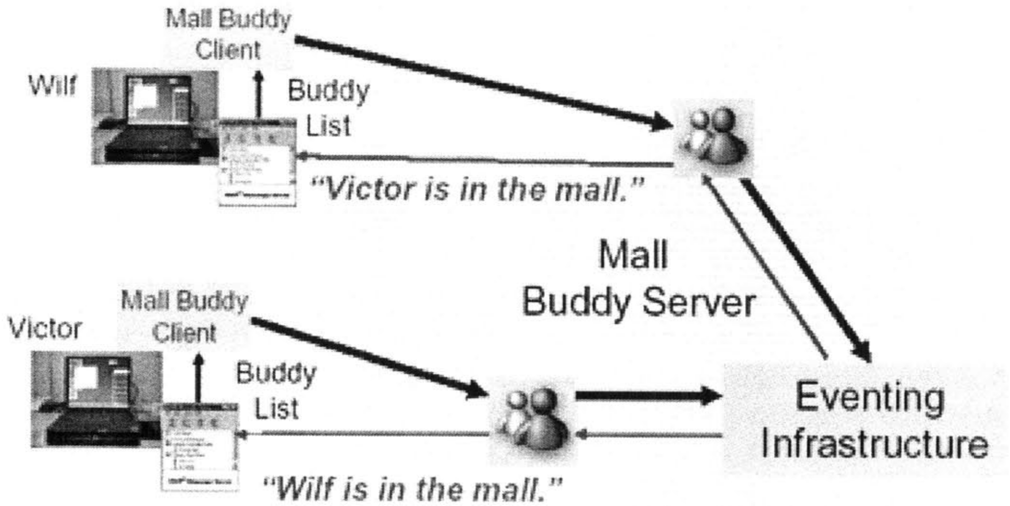


Figure 6.3: Location-Based Buddy List Service Architecture (Image Source: Victor Bahl, Microsoft)

One event source can generate trigger events of one or more event classes. A location trigger class has typed attributes, instances of which uniquely identify a captured activity. For example, an location trigger source that provides information about the locations of users can offer monitoring facilities for the following class of event, which identifies that an entity has changed location in a specific domain:
LocationTrigger(Domain, Name, Type, Location)

Triggered LBS services are more similar to traditional applications, instead of the Web model. The protocol might http, but it requires that the positioning infrastructure provider (the MPC server) provide the information at the trigger intervals, and there are currently three ways to perform this task: using a special CORBA method, using a proprietary interface, or using HTTP POST.

6.4 Context-Aware LBS Services

Several dimensions of mobility provide a basis for characterizing LBS usage context[76]. These dimensions include: spatio-temporal context, environment context, personal context, task context, social context and information context. Location positioning can be an important input for determining context for context-aware computing services[77].

6.4.1 Service Use Case

The service use case is similar to the "location-aware" type presented above. The difference here is that context-aware LBS services not only function based on location triggers, but also on event triggers, such as "*an item is on sale*."

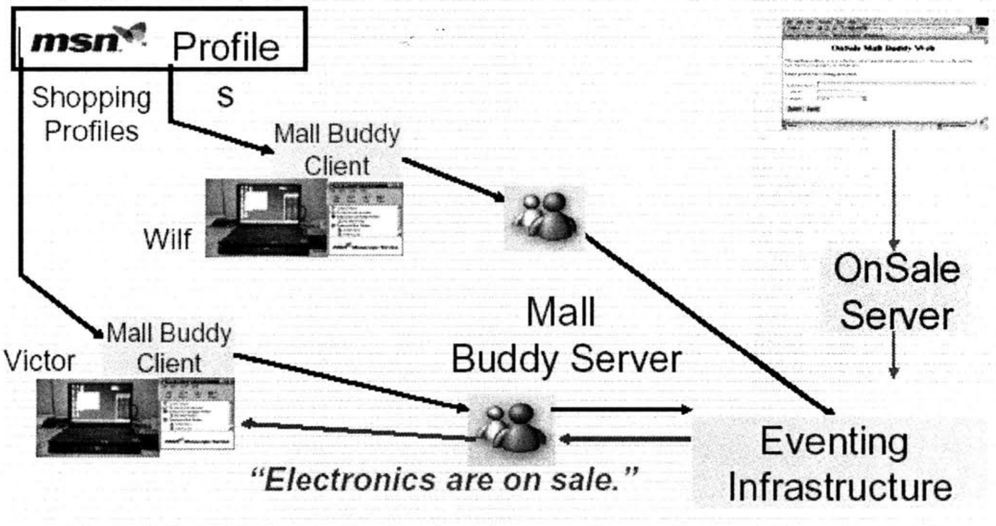


Figure 6.4: OnSale Mall Buddy Service Architecture (Image Source: Victor Bahl, Microsoft)

---

[76] Krogstie, J., Lyytinen, K., Opdahl, A., Pernici, B., Siau, K., and Smlolander, K.., "Mobile Information Systems - Research challenges on the conceptual and logical levels," presented at ER/ISIP 81. Workshop MobIMod'2002, Finland, 2002.

[77] Beigl, M., Zimmer, T., and Decker, C. "A location model for communicating and processing of context," Personal and Ubiquitious Computing, pp. 341-357, 2002.

6.5 “Push” versus “Pull”

From an architectural perspective LBS applications can be categorized as “push” or “pull,” which are two approaches to make the mobile device (or the service/network) aware of its current location.

In a “push” (passive) application, processing activity occurs only when it is initiated by the client. An example might be an application to find a nearby ATM where a user with a mobile device sends a query to a spatial server to return a map (or direction, or both) of the resulting ATM locations. Here, the mobile device passively listens to beckons from the cell base station. Moreover, the mobile device only queries a local database for location, it has complete privacy and it can choose to advertise its current location to the world or only to selected third parties.

In contrast, a “pull” application will send potential customers a text message with a special discount offer when they are close to a particular store. For this type of application to occur, the position of all users needs to be “pulled” or updated frequently and automatically. This is a real-time service, which may need to handle thousands of spatial updates per second.

6.6 Passive versus Active Systems

An example of an active system would be one that gathers data from APs – as they report measurements about RF transmitters between themselves and mobile devices – and uses this data to track these mobile devices. The system tracks the location by monitoring transmitters from mobile devices and the mobile device queries a central database to get the current location.

Simple mobile transmitters, combined with active LPS systems, are well suited for inventory management and programs to monitor and secure objects and assets. People, however, may not prefer to be tracked like objects or assets while roaming the halls and rooms of indoor buildings and campus structures. A compromised active LPS system could actually facilitate a data or even physical personal attack, as an intruder could watch – on-screen – from a distance and wait for a target to move to a vulnerable location. Plus, any residual tracking records from the software could be analyzed for patterns or other variables that infringe upon privacy norms and regulations. Active systems must be highly secured and precisely engineered in order to guarantee the safety and security of users. Further, from a practical perspective, pure active systems alone do not satisfactorily address the problem of disconnected operation, although they can provide high levels of positioning accuracy

A passive software LPS system is a functional as well as ethical alternative to active systems. In passive systems, the LPS protocol and data are publicly offered, and device-resident processes are responsible in the end for interpolating location. GPS is a passive system.

Architectural challenges associated with passive systems include optimum formats for data, synchronization and broadcast intervals, and logical signal differentiation – from the perspective of limited devices. A passive LPS system provides open infrastructure, anonymously broadcasting location-derivable information and inherently accommodating privacy.

6.7 The Software Platform

The indoor world is lacking a common platform that would integrate all the various components from each of the infrastructure types providers (from the communication infrastructure providers (i.e., cellular carriers), service and content providers, and software developers), who have been trying to identify the needs of a fully connected user, facilitating seamless LBS services. The results in the outdoor cellular scenario are already matured and location architectures have been standardized (i.e., OpenLS platform).

A software platform acts as a central system by gathering, integrating and transmitting data between a variety of different components, including the actual positioning technologies, the network, database servers, billing and service management systems, GIS servers, the end user's handset and the actual LBS applications. All of this is accomplished while implementing privacy rules that insure proper and authorized use of the end user's location information.

What is important to realize is that the indoor LBS applications reviewed in **Chapter 2** technology driven and written "bottom-up" to test or demonstrate the capabilities of a novel LPS system such as the ones reviewed in **Chapter 4** or to support further research. Applications developed this way lack portability due to reliance on intricate properties of the sensors, reference frames and location formats in use at the times of their development. This, in turn, will result in development difficulties based on assumptions about the underlying sensor technology. Consequently, replacing the sensor network must be accompanied by an overhaul of the application source code.

This platform, which we call '*Indoor-OpenLS Platform*' (**Figure 6.5**) needs to be able to connect to external information sources like the Internet using wireless communication and allow for "seamless handover" (**Chapter 4**). In the outdoor world, a wide area network (WAN) for data service of a cellular network systems (i.e., GSM, GPRS or UMTS) can be used. In the indoor world, a wireless LAN (e.g., IEEE 802.11) can be employed.

Moreover, this platform needs to integrate the different LPS systems and hide their heterogeneity with respect to positioning methods. With the ongoing research for indoor positioning mechanisms resulting in implementations that very in several factors and offer specialized APIs a common platform or framework is needed. This platform would enable the deployment of LBS in heterogenous positioning systems (i.e., Wi-Fi based) and would address difficulties cooperating with different systems' components.

Similar to the cellular network's approach with the GMLC/MPC standardized gateway[78] for location servers, the wLAN network should have its own standardized component, which we call the '*wLAN Location Center*' (WLC), to hide the heterogenous functions of the indoor positioning methods and architectures based on wLAN networks (i.e., 802.11 Wi-Fi) or Bluetooth. The WLC gateway, along with a LIF-like API would unify a framework for retrieving position data of users

---

[78] The European Telecommunication Standard Institute (ETSI) and the 3GPP specifications define the GMLC gateway entity which exists within the PLMN of the network operator.

from the LPS systems (as well as GPS and cellular network when considering "multi-worlds" and seamless handovers).

It is generally agreed within the industry that common abstractions (i.e., open service infrastructure) for designing these systems are needed. Openness is typically achieved by components having open (published) and standardized interfaces that support interaction through standardized and published APIs. We specifically focus on APIs like the LIF MLP and the OpenLS service APIs. **Figure 6.5** shows where these APIs are used for component integration and interaction.

Overall, we dissect the platform into five units:

1) Web client (i.e., user interface)
2) Application servers
3) Databases
4) Middleware
5) Positioning (and communication network) component
6) The communication protocols enabling cooperation between these units

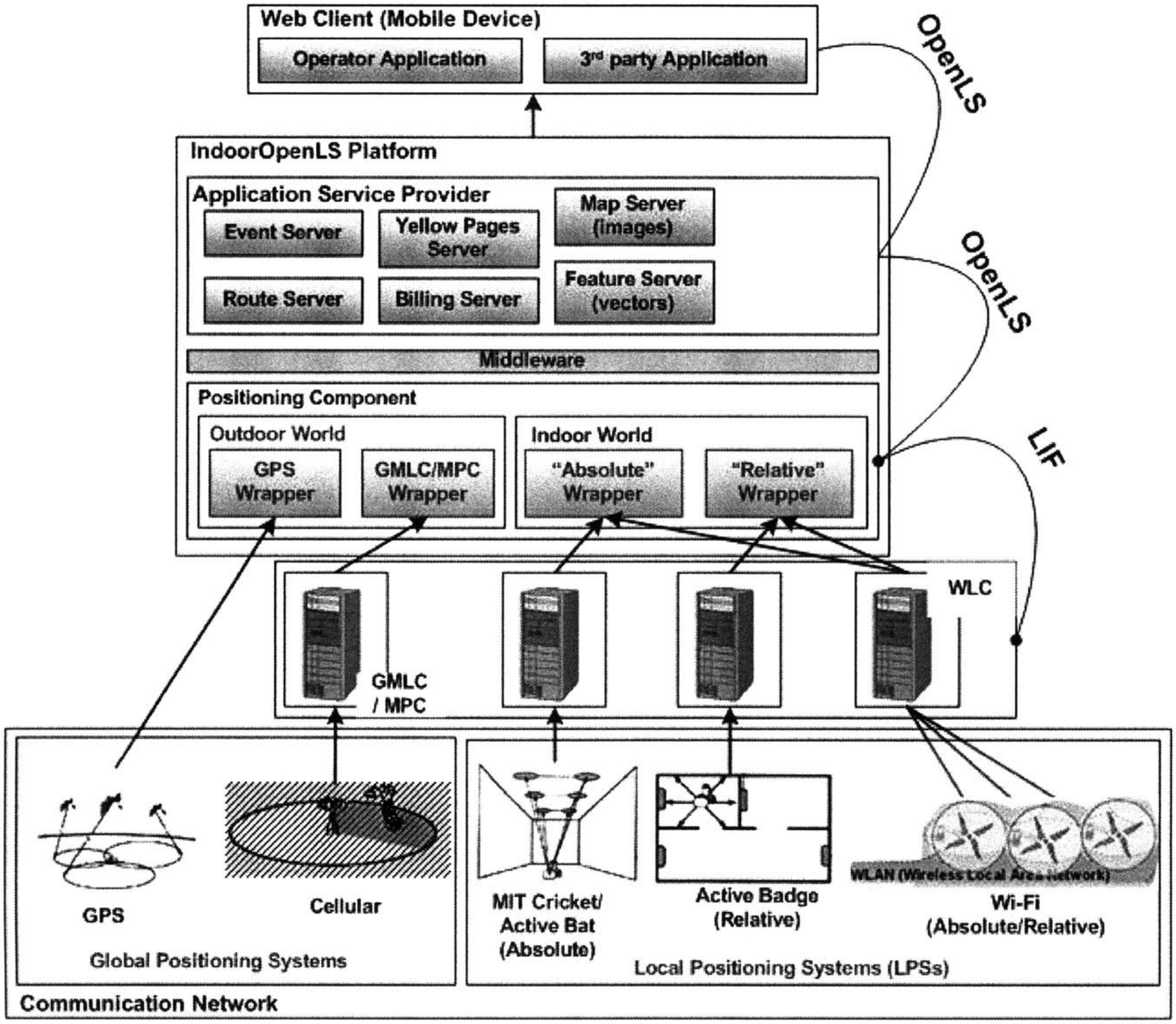


Figure 6.5: Indoor-OpenLS Platform/Framework

The platform should be designed to meet general industry agreed objectives such as modularity, scalability, extensibility, portability, and to facilitate open interfaces. The need for such a platform can be seen from the following example of launching and maintaining a compelling portfolio of location-based trigger application such as the Buddy Finder (refer back to **Chapter 2** for the use case). This task is challenging due to certain software and infrastructure limitations, which include:

- **Overloading the network**. Tracking a list of family/friends members in order to be alerted if they are in the vicinity would require the system to determine every 5-15 minutes if the wandering event has occurred. This type of load can overload the communication infrastructure's capability making the service cost-prohibitive.
- **Scalability of service**. Alert-based LBS involves continuously tracking mobile users. These services require dynamic time-tracking of users along a complex location model. In addition, the moving users have relationships (or triggers) to each other, other moving points and to static objects (i.e. "let me know when I am close to a friend"). Users may

also have complex relationships with these points or other users. For example, in a date-match finder service, a date has certain characteristics (age, education, hobbies, etc.). Adding more subscribers causes an exponential growth in processing required to manage these services. This computational intensity requires an a scalable system.

- **API to develop compelling services**. Application developers need an extensible interface for defining alert criteria and responding to alert events. This interface must support a variety of quality-of-service parameters that reflect the diverse nature of mobile location data in today's wireless networks, as well as a flexible privacy model for supporting a wide range of service requirements.
- **Customization of services and privacy by end users (user profiles)**. The power of alert-based LBS will drive consumers and enterprises to customize the settings and triggers of the services they subscribe to. For example, they may want to define trigger zones (location-based) or set times (temporal-based) when they want to be alerted of events and times when they don't.

6.7.1 Web Client (User Interface)

The Web client is about the user interface and how the user interacts with the mobile device / LBS service. This interaction pertains to the "simple" versus "location-aware" service use cases presented above. Other characterizations are connected with the client-side are the small screen, limited keyboard and limited bandwidth put premium on informative visuals, few keystrokes, heavy lifting at server.

The issue of mobile mapping is important in the context of LBS as the ability to display mapping information on mobile devices such as cell phones is limited. The location-based information may simply be text (i.e., point of interest), images, map images (i.e., area of interest), etc. In general, attractive maps are the best means of depicting location-based information, and are hence an essential element of LBS. Moreover, with the development of 3G telecommunications and the wireless Internet, users will be able to gain access to a wide variety of map information.

However, developing mapping applications for the mobile/wireless Internet is challenging, for several reasons. The major concern is the limited display capability of mobile devices. Apart from the limited map features that can be displayed, the speed of data transmission to mobile devices is also slow in comparison to a wired network. Moreover, each device speaks a different wireless protocol and supports a variety of different wireless markup languages – these different standards preclude a Web site developer from writing every application to individually support every single device available. For example, WAP-enabled cell phones support the WML. On the other hand, Palm Operating System devices support TTML (Tagged Text Markup Language), and voice-activated Internet applications support the VoiceXML and VoxML mark-up languages.

Mobile mapping requires standards that allow data content to be easily transferred and displayed across the wireless Internet to any one of a large variety of mobile devices. For displaying location-based data on the standard Internet, Scaleable Vector Graphics (SVG) and GML are two important standards. The data in these formats are delivered across the Internet via XML. In the case of the mobile/wireless Internet, there should be conversion of standard Internet markup languages (XML and HTML) to languages that mobile devices can understand (e.g. WML, Handheld Device Markup Language - HDML, VoiceXML and SMS).

One leading mobile middleware product that enables these conversions is the Oracle 9i Application Server Wireless Edition (Oracle 9iAS Wireless). It converts any Internet content to XML and transforms the XML to any markup language supported by any device (HTML, WML, HDML, VoiceXML, VoxML, SMS, etc.).

6.7.2 The Application Servers

Application servers are situated between the communication network, the databases, and the end user. These servers are essentially connection points for the different components of the software infrastructure. The most basic application server is the Web server, using HTTP to call the MPC and retrieve the position information (**Chapter 4**). In a Web environment, the application server requests a position by issuing an HTTP POST request towards the location server. The query is invoked by sending the request using HTTP POST to a URI, which is used to transport the data. XML is used to encapsulate the requests and responses and have functions to report the quality of the position.

In addition to incorporating Web servers, application servers also provide database interfaces (using CORBA, XML, EJB, JDBC, ODBC interfaces). The Web server usually acts as the front end to the application server, using the mechanisms of the Web (http) and WAP (html, wml, xhtml) to deliver the presentation. The query by a user can combine multiple data requests, such as the subscriber's location, content information of multiple kinds, and mapping information of any kind.

The data flow is managed differently depending on the application server provider, and architectures. Sometimes the application server is housed in a central location, like an ISP, and the service providers can house their applications in it, and it takes care of the interfaces to the network. Some of the network operators who have deployed LBS services see themselves as ASPs; others plan to sell the data to companies that want to provide services. The services would then be provided in the same way as Web service today – by companies that essentially are publishers but run their own infrastructure.

For an efficient communication within a network, the amount of data that has to be transferred between Web client and the application servers must be minimized. The efficiency of data transmission must be increased by means of caching or hoarding techniques. For this reason, the application server should preserve the states of its location aware clients (e.g. containing the map extent or the specification of results of previous queries). To minimize location signaling overheads and to avoid power consumption of mobile devices, the location cache would keep the location information of the recently tracked user. Nibble for example, a Wi-Fi based location system, introduced a predefined refresh period, which imposes lower bounds on the 'time to respond' requirement.

Some application servers provide for personalization beyond the use of position information as well as the transcoding (i.e., using XSLT) of information. But neither of these standards, and we will not discuss them in any depth – although it is clear that personalization services are essential to the future of LBS services.

GIS Application Servers

Spatial (GIS) servers provide data about static or mobile objects and users, and upon a request, return their location. They also provide the necessary geoprocessing functionalities (i.e., shortest path calculation). Each GIS server stores information about spatial objects within a particular area. For example there could be a GIS server that stores 2D and 3D shapes of buildings. Distributed servers collaborate to provide a unified spatial view through a defined API to applications. At the end of this chapter we focus on the OpenGIS Web Services APIs and the OpenLS APIs. In addition, it is important that the servers are able to describe locations at different levels of precision and scale (refer back to **Chapter 5**).

To achieve the scalability necessary for a large-scale deployment of the platform, the GIS servers can be organized in a hierarchical fashion, similar to that of the Globe location system for software objects[79]. Leaf servers in this hierarchy are responsible for managing the position and registration information for the mobile objects inside their disjunctive service areas, while the higher level servers are responsible for forwarding queries and handovers.

As a fast processing of queries and especially updates concerning location information is of great importance, the location information can be managed in a special main memory data structure based on a Quad-tree[80], while the registration information is stored in a traditional database. The volatile position information, which will be out-of-date after a server failure anyway, can be recovered from the mobile objects. This is the approach that the NEXUS platform took.

A special kind of a GIS server is a route server that computes a path through a path network, given two or more positions. A typical application of the Route Server is the ability to calculate and display the best or shortest route between two specified points on a path network. However, for the indoor world, where relative positioning (**Chapter 4**) and symbolic location data models (**Chapter 5**) seem to be the norm, such GIS servers need to be adopted for such purposes.

6.7.3 The Positioning (Communication) Component

As we described in **Chapter 4**, the position information can be derived from the positioning infrastructure by using an API such as the LIF MLP API. The application server essentially works as a gateway between the MPC (or the WLC) and the mobile device. That is the way it is intended to work in the LIF API. In theory, the mobile devices can connect to the application server by using any protocol: DNS, SMTP, or FTP. It functions as a gateway toward the LPS system. With the LIF API, the application server interfaces to the mobile cellular network of the mobile operator in LPS systems (such as assisted GPS and network-based positioning). The API is used to call the position information, which means that developers don't need to know the

---

[79] Steen, M. v., Hauck, F., Homburg, P., and Tanenbaum, A.: Locating Objects in Wide-Area Systems, IEEE Communications Magazine, pp. 2–7, 1998.

[80] Samet, H.: The Design and Analysis of Spatial Data Structures. Addison-Wesley, Reading, MA, 1990, ISBN 0-201-50255-0.

backend implementation details of the operator/carrier (nor don't have to write applications that interface directly to the system of the operator). Querying the proper MPC or WLC retrieves the user location.

A LPS system providing an absolute position can usually be augmented to provide corresponding symbolic location information with additional information. For example, a Web client application can access a separate database that contains the positions and geometric service regions of other objects to provide applications with symbolic information. Applications can thus link the physical position to determine a range of symbolic information like getting to know the closest printer to the current position.

6.7.4 The Databases

The positioning information that is obtained from the LPS systems (**Chapter 4**) is managed within databases, from which the application server will access the data for various LBS services. RDBMS are generally used to store location data. However, a RDBMS is designed only for transactions involving comparatively simple data types such as characters and numeric data. Location data, especially data based on the geometric location model (**Chapter 5**) are usually complex objects that require more than one data structure to describe them, and their spatial relationships (i.e., topology).

Object-oriented RDBMS merges the object-oriented management system that allows the storage of complex data as objects, and the relational database management system to offer the ability to manage the relationships between objects. A Structure Query Language (e.g. SQL2 or SQL3) can support all database management operations, as well as object-oriented data modeling. These enable users to store and manage complex location data, as an object, along with data from other sources such as CAD and images in the same database. More importantly, ORDBMS allows spatial analysis to be performed in the database server using SQL commands instead of in the application. Oracle8 Spatial and ESRI ArcSDE are an example of spatial databases that stores geometric objects as Abstract Data Types (ADT), a user-defined data type, in feature-based tables, within the RDBMS.

From a GIS perspective, LBS services do not include many complex spatial analyses. However, it is the nature, completeness and accuracy of the database content that impacts on the quality of the subsequent LBS service. For a certain service area, the database must include all the appropriate features such as hallways and points of interest (POIs) such as ATMs, restaurants, stores, etc. In addition, digital 'maps' of the area are needed. These can be a portfolio of raster maps (images), a vector map that can be created 'on-the-fly' when requested, or archived photographs. All hallways and points of interest (and appropriate labels) must be shown, and be geo-referenced so that its location on the 'map' is correct. Nonetheless, applications that run on top of a geometric location models (**Chapter 5**), the spatial data analysis might require geometric functions involving the computation of distance, area, volume and directions.

With respect to the real-time nature of LBS applications, as more devices and applications become available, the increase in location-sensitive data requests will skyrocket. Communication infrastructure providers (i.e., wireless carriers, Wi-Fi network providers) will need to support this increasing number of moving subscribers that might be well beyond the amount of 20,000 customers per processor that a traditional system scales to. Database vendors obviously have great experience in handling high volumes of short transactions, so the trend toward more real-time applications is likely to reinforce the growing role that the major database vendors have been playing in the GIS market.

The traditional approach to analyzing real-time data is to continuously load the new information into a database management system and repeatedly run queries against the data, but there is significant overhead in indexing such dynamic results. A new approach to approach to real-time data analysis that is complementary to existing database technology would be valuable. A company called Apama[81] has developed a approach where it indexes the queries using the analytical model. Such queries can change over time, but they are much less dynamic than the data. Incoming data can be efficiently matched against relevant queries and acted on as appropriate. Also, as such applications grow and update rates grow to thousands per second, some potential niches open for new technology approaches. A company called Wavemarket[82] has developed an in-memory spatial database. Also, Oracle[83] now provides some in-memory spatial processing capabilities in its application server.

Other databases can be the user and security database. A user database stores the information that is relative to the registered subscriber. Each user is assigned a unique id. Just like with the cellular network's HLR and VLR database, user can be classified as 'visitors' or 'home' (permanent) users. Moreover, this database is used for accounting and charging purposes, including post and prepaid options. A security database holds all the required information that enables authentication and security to enforce the policy for the platform (WLC gateway)[84].

6.7.5 The Middleware

The central task of a platform deals with the data management. The platform should adopt a scalable middleware (connector) that is capable of handling huge amounts of data. To achieve this, the system has to be organized within a distributed environment. The middleware is the central entity within the platform that incorporates all the logic that is required to coordinate the other components. It is responsible for the data exchange between the different components of the platform. Generally, the indoor world environment consists of different networks and so the communication handover between them has to be solved as we explained in **Chapter 4**.

Moreover, the platform should consist of a middleware that will determine the positioning mechanism and the LPS system that is appropriate to serve the location request. In a Wi-Fi network, for example, the WCL gateway might be deployed in an environment where various LPS systems and architectures might coexist offering different type of service in terms of accuracy (relative versus absolute positioning).

---

[81] Apama: http://www.apama.com

[82] Wavemarket: http://www.wavemarket.com

[83] Oracle: http://oracle.com/

[84] In the cellular network, the GMLC gateway is responsible to provide the location information of subscribers that have been registered to the PLMN network and have decided to permit their positioning.

From previous work, the Spatially Indexed Resource Identification and Tracking (*SPIRIT*) system was an event-driven middleware_developed at AT&T Laboratories Cambridge. The *SPIRIT* project designed and implemented a three-tier high-level software architecture over the raw co-ordinate positions generated by their proprietary Active Bat system. The software performs the various mechanics of 2D spatial data handling and frees the application programmer from repeated re-implementation of such. Although the design goals included the ability to reason about a wide range of sensor technologies the software authors admit they never came close to realizing this objective and *SPIRIT* is closely coupled to Active Bat. The most significant outcome of the project was the '*Programming with Space*' API that has proved to be extremely well suited to the task of developing sentient applications.

6.7.6 Communication among the Components

In order for the platform to receive requests and to send responses from the web client (mobile device), the syntax should use open interoperability standards like the OSA/Parlay API and the LIF MLP API. Nevertheless, a proprietary legacy interface should be easily integrated into the platform.

As mentioned in **Chapter 5**, the Augmented World Querying Language (AWQL) is an XML-based query language wrapped in a SOAP[85] message and sent using the http protocol. An http server passes the request to the AWQL servlet that first uses an XML parser to extract the query from the SOAP message. The AWQL request is then passed to the AWQL parser component that ensures the validity of the enquiry and converts it to a sensor request. The sensor request is passed to the Sensor Worker component, which is able to access the sensors needed for answering the request. The AWML composer receives the sensor readings from the Sensor Worker and returns an AWML message wrapped in a SOAP envelope to the servlet. The servlet then passes the response to the requesting server. This query request-response cycle is illustrated in **Figure 6.6**.

---

[85] SOAP [0]is an Internet communication protocol that is independent of the underlying transport protocol and programming language.[0]

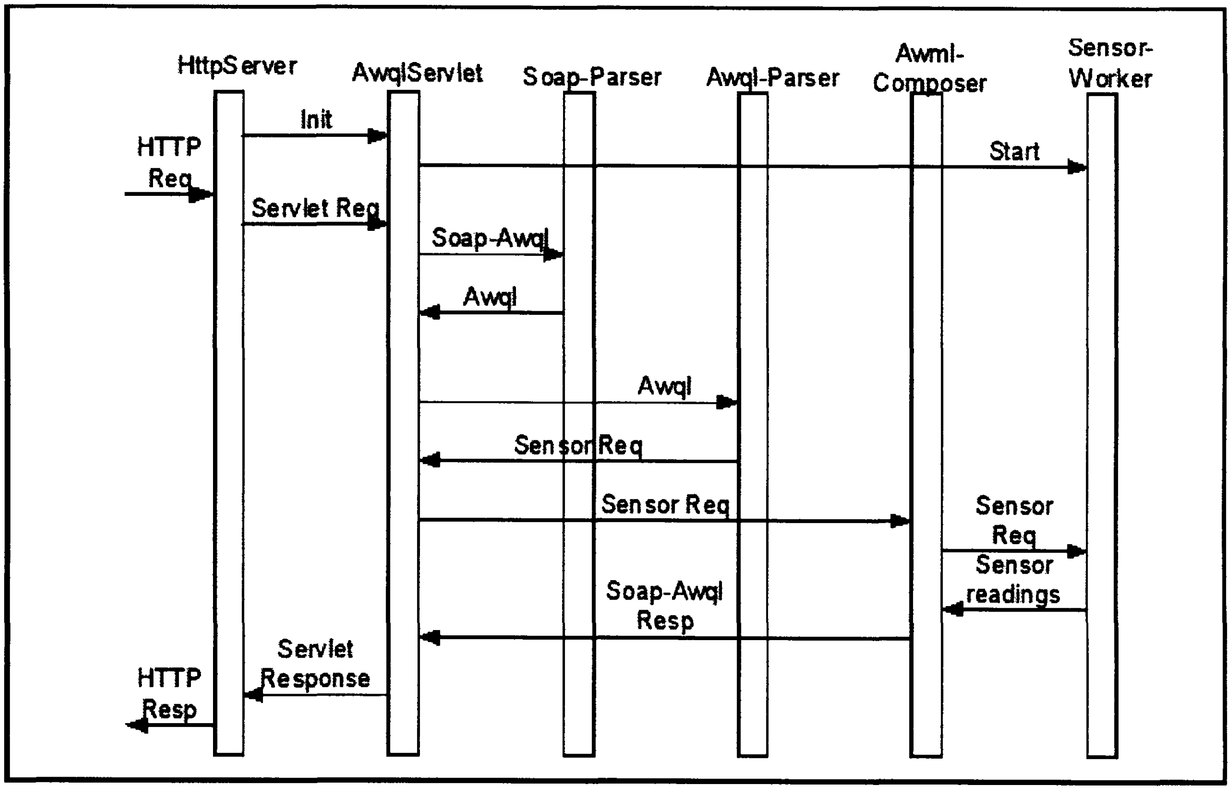


Figure 6.6: AMQL Query Process (Image Source: NEXUS)

6.8 Web Services

In a Web services model, application developers build LBS applications by using a suite of enabling technologies and APIs. They then publish these applications back to the communication infrastructure provider (i.e., wireless carrier), and this provider then makes the applications available for subscriber use. This enables Web client applications to perform spatial queries without knowledge about the respective servers.

Multiple dynamic devices motivate the need for the separation of hardware device control, internal computational logic and user interface presentation. Rather than tightly coupling input/output devices to applications, it should be possible to flexibly change the interaction mechanism without requiring modification of the underlying application. Web service enable decoupling and to flexibly exchange data between applications.

6.8.1 Service Discovery and Metadata

Service discovery is essential to indoor mobile users, providing the means to exploit services that are offered and to configure end-devices automatically and to register on LBS applications with the minimum (or zero) user intervention. This mobile environment must support a changing collection of services as the user moves within the "multi-world" environment. As a result, the platform should adopt the Web services' model (see **Figure 6.7**) of service discovery (find) and to advertise (publish) the LBS services that are offered in a particular area/region and manipulate the service discovery (bind) and the service configuration for end users.

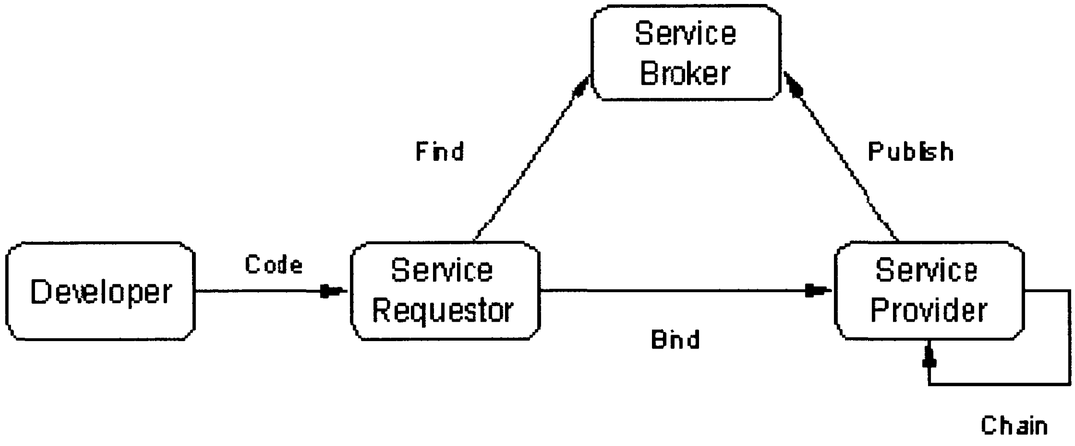


Figure 6.7: Web Service Publish-Find-Bind Model

First, the mobile application must discover the existence of newly available services. MIT Cricket, for example, allows applications running on mobile and static nodes learn about services in their vicinity via its Floorplan ActiveMap application that is sent from a map server application. It interacts with services by constructing queries for services at a required location. Once location information is obtained, services advertise themselves to a resource discovery service such as the MIT Intentional Naming System (INS)[86], IETF Service Location Protocol[87], Berkeley Service Discovery Service[88], or Sun's Jini discovery service[89]. Another example of such service discovery is Microsoft InConcert's lookup capabilities in the MS EasyLiving system[90].

Moreover, service discovery platforms such as SLP[91], UPnP[92], Salutation[93], and UDDI[94] attempt to move up from purely network-based addressing, to account for higher-level descriptions of networkable entities. They provide a bootstrap mechanism enabling the dynamic, spontaneous hookup of services and devices in ubiquitous computer environments. They use, in a centralized or distributed fashion, a generalized lookup service that may build upon and subsume the more specialized naming, trading or directory services provided by underlying middlewares and protocols.

---

[86] Adjie-Winot, W., Schwartz, E., Balakrishnan, H., and Lilley, J. The design and implementation of an intentional naming system. In Proc. ACM Symposium on Operating Systems Principles (Kiawah Island, SC, Dec. 1999), pp. 186–201.

[87] Veizades, J., Guttman, E., Perkins, C., and Kaplan, S. *Service Location Protocol*, June 1997.

[88] Czerwinski, S., Zhao, B., Hodes, T., Joseph, A., and Katz, R. An Architecture for a Secure Service Discovery Service. In Proc. 5th ACM MOBICOM Conf. (Seattle, WA, Aug. 1999), pp. 24–35.

[89] Jini (TM). http://java.sun.com/products/jini/

[90] MS Easy Living project: http://research.microsoft.com/easyliving/

[91] Srvloc: www.srvloc.org

[92] UPNP Org: www.upnp.org

[93] Salutation Org: www.salutation.org

[94] UDDI: http://www.uddi.org

Distributed directories also apply in the context of service directories. Distributed naming and directory services, such as the Internet domain service (DNS)[95] or the X.500[96] directory service offer scalable and fault tolerant design to provide directory service to a very large number of clients. Mapping names to addresses is not much different from mapping names to locations. However, directory services such as DNS do not cope very well with frequent updates. As a result, the requirement for real-time information delivery is not met by those designs. On the other hand, service trading, a special kind of directory function, is required to cope with functionality changing information.

Next, the mobile application must determine the newly found service's capabilities. Descriptions of services in the EasyLiving system are accomplished using a simple, open XML schema. In addition to ease of use, XML was chosen for two reasons. First, Extended Stylesheet Language (XSL) provides the ability to translate XML documents into multiple layouts. Second, it is straightforward to transform an XML-encoded description of a command into the XML-encoded command to be sent to the service. The service description schema is designed to support queries about available commands and their legal values. Additionally, the commands are associated with human-readable tags. While not a complete solution, this is a first step toward the automatic generation of user interfaces for different modalities.

As the number of services and catalogs available in an environment grows, there will be an increasing need for more sophisticated search-engine-like tools that can consolidate, organize and present information retrieved from various sources (see **Figure 6.8**). Such tools may also provide interfaces through which users can pick services they need. Such tools can dispatch the users' requests to a variety of available catalogs, and then allow users to sort the results according to different criteria (e.g., store location, price of product, quality or provider). As such, these tools are similar to popular online price comparison sites (e.g., metaprices.com or simon.com) which allow users to pick a category of items to compare (e.g., cds, books, electronics) and then return a list of items along with their prices, availability, special offers and reviews from various online shopping websites.

---

[95] DNS: http://www.dns.net/dnsrd/

[96] X.500: http://www.isi.salford.ac.uk/staff/dwc/Version.Web/Contents.htm

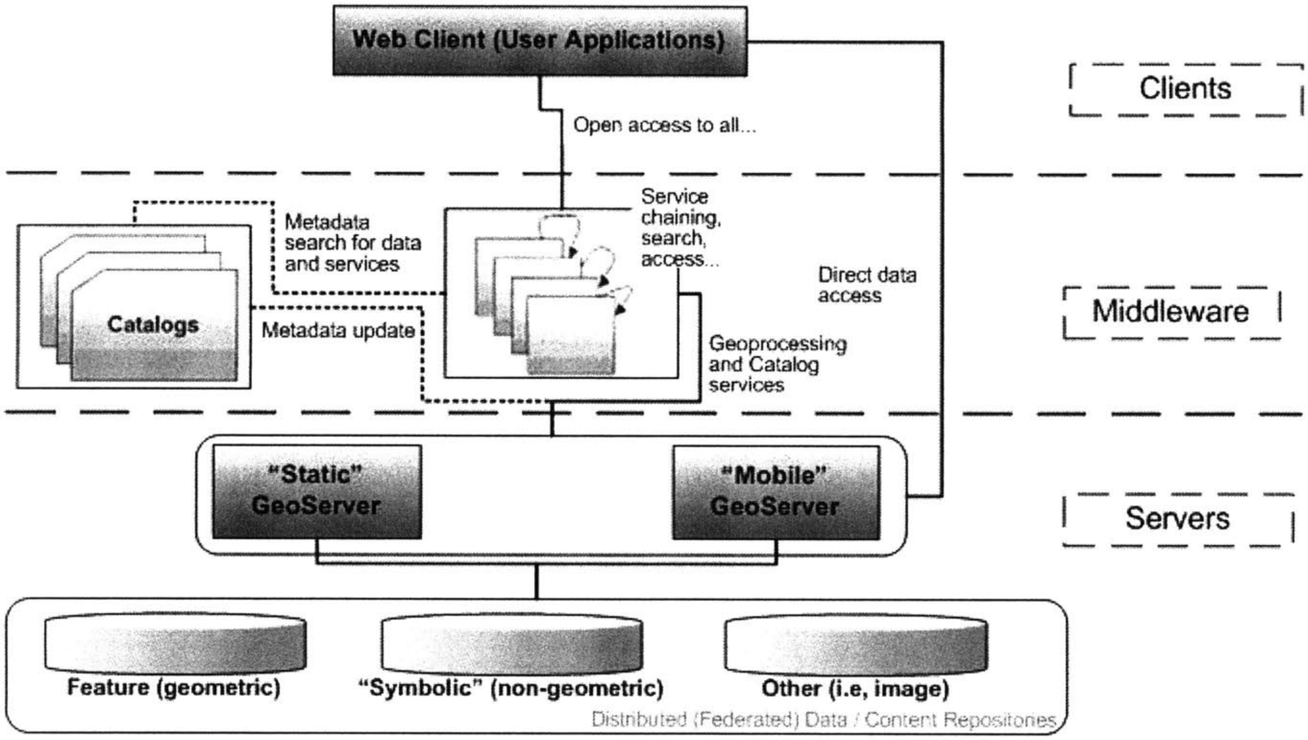


Figure 6.8 Service Catalogs

Service-bundling has the potential to provide huge market entry opportunities for new players. See **Chapter 7,** where we explore different scenarios, including entry strategies for startup firms.

6.9 The Role of Open Interoperability Standards

The objective of such a common platform is to provide a foundation that makes the development of LBS applications much easier and allows for a better integration of and interaction between different applications. The technical environment in which location platforms operate has grown considerably more complex, with new standards bodies and new application areas continuing to appear. For the industry to fully exploit the revenue potential of LBS services as well as provide all the related management and support functions, a wide range of open and standardized APIs and features must be supported for a seamless a way as possible.

6.9.1 LIF MLP API

As we discussed in **Chapter 4**, the LIF MLP API is used to seamlessly integrate location from the communication network that is position-enabled (i.e., GSM, Wi-Fi cell-id based positioning). Specifically, this API servers as the interface between a location server and a location enabling server, which in turn is interfacing with the application server.

The API describes the request/response that gathers position information (i.e., *x,y* coordinates) from the MPC/GMLC servers. Moreover, it defines the core set of operations that a location server should be able to perform. The location-enabling server (middleware) effectively acts as an LIF-MLP pass-through and subsequently passes on requests from the LBS application to the MPC/GMLC for location gathering. There is also a function to set a privacy flag in the HLR database. If the flag is set, the user has requested not to be positioned, and it can be overridden only by an emergency request.

There are two main functions for retrieving location information: immediate and deferred (for example, triggered by a timer). The LIF API assumes that the location information is delivered to the client (or the application server) as a result of a query. The GEO_INFO element stores the position information. Position can be in xy coordinates, for example.

The API is formally defined in a number of XML DTDs. They give the type definitions for XML elements that are to be sent in the different documents that comprise the messages. Because there are a number of common structured elements among different services, the DTD that defines a single location query service is composed only by the definition of the root element and the inclusion of the necessary common DTD. In effect, the DTDs define data structures for the HTTP methods.

6.9.2 OpenGIS Web Services (OWS)

Enabled by the advancements in Web services in general, the OpenGIS Web Services architecture is rapidly manifesting itself. Various groups within OGC are working on service categorization (data, processing and registry/catalog services), data encodings (SLD[97], GML), and service chaining (WMS/SLD/CPS), which, overall, is setting a precedent for service chaining in the Web services environment. Within this work, general Web services technologies have been critical: examples include WSDL for service description, UDDI for service discovery, SOAP for passing XML-encoded data, and IBM WSFL and MS XLANG for Web service composition and process languages for orchestrating web services (OWS1 2002[98]).

The WMS Interface

In particular, the OpenGIS Web Map Server (WMS) specification[99] explains how a map-image server should respond to basic queries such as GetCapabilities (tell the client application what the server can do) and GetMap (send back the requested map in the format solicited). WMS enables programmers to add interoperability to their geoprocessing systems. Using *http*, the tranfer protocol used by the Web itself, the specification defines a request and response protocol for Web-based client/map server interactions.

Web client support for WMS, in simple terms, means with just a URL of a map server that also supports WMS, your desktop or Web application can tap into layers of data from any WMS-compliant servers, no matter what software is being run. Technically, the WMS specification determines how the client and the server communicate about what data is available, how it's delivered and how information about map features is delivered. But to use WMS, you just need a URL.

---

[97] OpenGIS Style Layer Descriptor (SLD) Specification: http://www.opengis.org/techno/discussions.htm

[98] OGC Web Services Initiative 1 (OWS1) Baseline Documents Page: http://ip.opengis.org/ows1/docIndex.html

[99] OpenGIS WMS specification: http://www.opengis.org/docs/01-068r2.pdf

The publication of the WMS specification has caused traditional GIS vendors to "wrap" their proprietary server products, adding OpenGIS interface software which allows them to respond to native (proprietary) and WMS (open) queries.

The WFS Interface

Similarly, the WFS specification[100] focuses almost exclusively on data encoding and transmission. Moreover, how the datasets served up are to be displayed and used is left to developers of Web client applications.

WFS describes data manipulation operations on OpenGIS Simple Features such that servers and clients can communicate at the feature level. Therefore, a Web Feature Server request – like those supported in many GIS and RDBMS packages – consists of a description of the query and data transformation operations that are to be applied to WFS enabled spatial data warehouses on the Web. The request is generated on the client and is posted to a WFS server. The WFS Server "reads" and executes the request returned in a feature set as GML. A GML enabled client then can use the feature set.

Therefore, whereas WMS delivers a picture, WFS implemented in a client supports the dynamic exploitation and access of feature data and associated attributes on the Web from any server product that implements WFS. This capability opens the door to enhanced spatial analysis, modeling and other operations based on the intelligence of the attributed data. Beyond feature access, there is an additional set of interfaces in the WFS for supporting simple transactions on a feature: *Create, D*elete, and *Update.*

The OpenGIS Catalog Service

The OpenGIS Catalog Service Interface Specification[101] defines a common interface that enables diverse but conformant applications to perform discovery, browse and query operations against distributed and potentially heterogeneous catalog servers. Spatial Catalog servers typically contain metadata about spatially referenced information such as maps, schematics, diagrams, or textual documents. The specification uses metadata and spatial location to identify and select layers of interest, and provides for interoperability in catalog update, maintenance, and other Librarian functions. The specification is designed for catalogs of imagery, GI, and mixtures of the two. It specifies open APIs that provide discovery services, access services and interfaces for catalog managers, including a complete Catalog Query Language.

Detailed implementation guidance is provided for establishing and ending a stateful catalog session to: query the catalog server properties, check the status of a request, cancel a request, issue a query, present the query results, and get the schema of a discovered collection.

---

[100] OpenGIS WFS Specification: http://feature.opengis.org/members/archive/arch01/01-023r1.pdf

[101] OpenGIS Catalog Service Specification: http://www.opengis.org/techno/specs/99-051.pdf

6.9.3 OpenGIS Filter Encoding

To achieve application-independent location data, and thus location-independent location-based applications, several conventions will be required:

- An application-independent interface for querying data/service ;
- A location-authority-independent representation of "a location" as a data object;
- A "universal federation" in which all location models combine to create a "virtual location model" to cover the (indoor) world in one framework.

In addition to GML, which we discussed in **Chapter 5**, the OpenGIS Filter Encoding serves as a common data API that provides the functionality needed for typical location-aware applications and hides the details of the underlying data management. Queries to this interface should be formulated using an XML-based language, which contains the following elements:

(a) a restriction, which is a boolean expression made of relations between attributes of objects and fixed values,
(b) a filter, which allows an application to remove attributes it is not interested in from the result, and
(c) a closest-predicate, which allows to narrow the result to the *n* objects closest to a given position.

6.9.4 The OpenLS Platform

Part of the OpenLS Platform is the location enabling server, which is a middleware module. The software interfaces to the MPC/GMLC and all other carrier/operator systems (e.g., WAP-GW, SMSC/MMSC, OAM&P, and billing systems). This centralized interfacing minimizes integration efforts each time a carrier/operator launches a new LBS application.

Perhaps more important, this location-enabling middleware also handles privacy, presence, and personalization, which are absolute must-haves for LBS, especially in the vertical mass market. Finally, it interfaces to a GIS engine, commonly referred to as a GeoServer. In some cases, the GIS engine is considered a transparent layered function of the location enabling middleware.

There are several commercial offerings for location-enabling middleware. An example is the ESRI's ArcLocation Solutions that consists of a spatial server (GeoServer) built on top of industry-standard SOAP and the OGC's OpenLS XML API suites. This middleware handles services chaining, MPC/GMLC integration, and mobile application server integration. It also consists of toolkits for third party developers.

The OpenLS GeoMobility Server architecture is an open location services platform. It uses OpenLS interfaces to access network location capacity (provided through a GMLC, for instance) and provides a set of interfaces allowing applications hosted on this server, or in another server, to access the OpenLS Core Services.

The OpenLS APIs are interfaces (XML schemas) are for implementing interoperable LBS applications, and are used for accessing directory services (such as yellow pages), route determination, location determination gateway, geocoding, reverse geocoding, and portrayal services using standard Web protocols. The APIs allow telecommunications companies, traditional GIS technology companies, and LBS application providers to efficiently implement interoperable LBS applications that seamlessly access multiple content repositories and link with other LBS services. They also enable seamless communication handover among cellular wireless communication networks (and devices).

The OpenLS objective is to provide a standardized solution for carriers/operators that allows them to choose and implement standard interfaces and components into their LBS systems, ensuring that application developers have standard tools and/or services to use when building LBS applications. The OpenLS Navigation Service, for example, determines routes between two or more (outdoor) points having specific (x,y) coordinates.

As shown in **Figure 6.9**, within the OGC Services Framework, there is a range of services and content protocols, which include:

- *Location content access services* (i.e., Web Map Service, Web Feature Service) these provide access to repositories of geospatial data.
- *Geocode, geoparse and gazetteer services* – determine the geographic location for addresses (a store inside a shopping mall), landmarks (i.e. ATMs), places and other textual/coded location descriptors.
- *Coordinate transformation services* - these provide the coordinate transformations between various projection coordinate systems.
- *Discovery of location services and content holdings* (i.e., basic service model and catalog services) - used to discover location services and location content.
- *Portrayal services* (i.e. Style Layer Descriptor (SLD) and Legend) - provide for the customization, tailoring and understanding of the display of geospatial information.
- *Location content encoding and transport protocols* (i.e., Geographic Markup Language (GML), Location Organizer Folder (LOF) and Geolink) - these content specifications apply to the encoding and transport of collections of related location content.

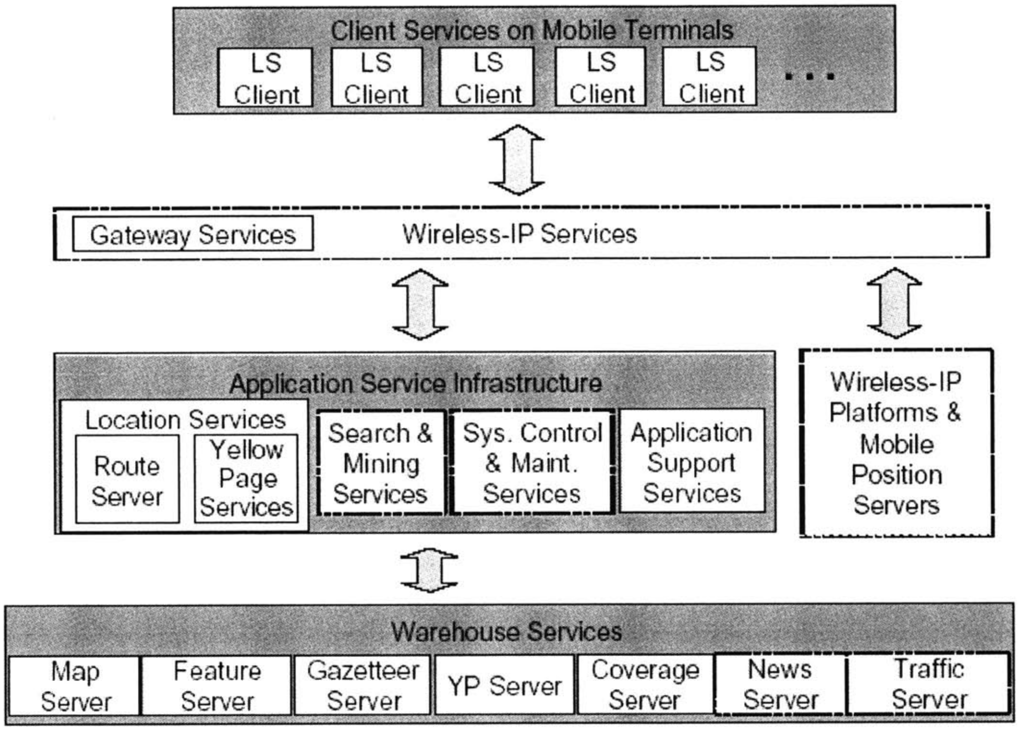


Figure 6.9: OpenLS Location Services Framework (Image Source: OGC)

6.10 Conclusion

In this chapter, we considered technological driving forces such as Web service to shed some light on how these standalone applications can be integrated by utilizing service-chaining and open interoperability standards. We then explored OGC's OWS and OpenLS, and outlined which components need to be standardized to enable interoperability. This, in turn, will accelerate the deployment of any kind of LBS service. In the absence of open and standard APIs provided by the various vendors and to the various services, on the other hand, integration or the bundling of such services would take extra time, effort, and requires specialized knowledge.

The 'indoor world' can learn from the 'outdoor world' and start applying the accepted standards for the appropriate software components as an early adopter. Of course, incremental growth needs to be considered and standards need to be studied first to see how they fit to specific needs. Trying to do too much coordination and utilizing standards to achieve interoperability too soon might not be economical (using cumbersome database management packages, handling many users or too much detail, worrying about editing and updating, trying to keep all the evolving interface standards in sink, etc.)

However, the providers do recognize that sooner than later they would like to use their services in other settings so they have an interest in standards and integration actions that facilitate coherence and long-term goals while being easy to do and inexpensive (in time and money and performance) in the short run. In short, the 'indoor world' might decide to apply the outdoor world standards, try to modify them, or design new ones. Of course, preference would be to apply them. These scenarios/strategies are explored next, in **Chapter 7**.

Furthermore, as the market for indoor LBS develops, many companies may want to deploy multiple vertical applications on top of a common software infrastructure. For example a hospital may want applications for security, office productivity and intelligent building infrastructure, in addition to specific healthcare applications. It is important that the overall platform can handle multiple applications developed by different vendors or organizations, running concurrently.

CHAPTER 7

SCENARIOS: BUSINESS AND TECHNOLOGICAL STRATEGIES FOR THE INDOOR LBS MARKET

7.1 Introduction

As mentioned in **Chapter 1**, the indoor LBS market is in its very early stages. As the indoor LBS market becomes more mature, there will be an increasing need for collaboration among the different market players. The purpose of this chapter is to provide a basis for discussing market developments and the places where open interoperability standards can/should make a difference, be mutually beneficial or require user pressure or new partnerships.

As part of the scenario development, the goal is to explain which issues are more important, what are the key choices, tradeoffs, and decisions that impact the timing and practicality of integration and interoperability. In addition, in terms of the various components in each of the infrastructure types, this chapter explores what buildup, incremental steps, fragmentation are useful and/or likely to happen.

Companies competing in the marketplace can capture value through a variety of strategies, which are explored in this chapter. Although some may exercise the advantage of being first-movers, the benefit of having tight appropriability on their technologies and applications may not be enough and might lead them into the development of complementary assets and common open interoperability standards to gain market power and extract rent.

Before getting into the scenario analysis, it is worth to come up with some questions to frame our way of thinking. In **Chapter 6**, we outlined some questions that deal with the technical integration of the various components from each of the infrastructure types (and from different providers). Here, we list those that deal with the business complexities of effectively delivering indoor LBS services:

- Who actually sells the service?
- How does a bundled service adhere to industry regulations/standards? (business viewpoint)
- How do consumers subscribe?
- How do bundled services affect competitiveness in the market?
- How do you charge for the service and distribute the income between the constituent providers?
- Can you assure a common quality of service level between the constituent services?

The following analysis will address each of these questions to some degree that will outline the choices, tradeoffs, and decisions that should be considered when playing within the LBS value-chain.

7.2 The Scenarios and Strategies

The purpose of Scenario 1 is to explore a roadmap for a "Stovepipe" (or "self-contained") company such as MIT, and to construct two possible business paths or strategies that such a company could follow. Currently, MIT is a provider in all of the infrastructure types. The two strategies that MIT can follow are: (1) an end-to-end solution provider; and (2) an incremental growth approach where MIT would specialize and/or outsource.

The purpose of Scenario 2 is to explore a roadmap for a company that specializes in one of the infrastructure types, such as the software infrastructure, and to construct three separate business strategies that such a company could follow. Even more so, this company (referred to as "Startup") will provide a subset of the software infrastructure type. **Table 7.2** shows that these subsets can include the software platform, application development, and service integrators. We chose the Startup to specialize in application development, such as the Buddy Finder. The three strategies for the Startup relate to competing against big players, specifically wireless carriers. Moreover, these are entry strategies that companies in the industry can target to compete with wireless carriers.

Since a wireless carrier is of such large significance in the overall LBS value-chain (both for the outdoor and indoor world), we believe that the current indoor "Stovepipes" and "Startups" companies will need to eventually establish partnerships with these predominant players in order to expand beyond their niche markets/spaces. The wireless carriers have a strong relationship with the wireless customer, giving them a natural leadership position in this emerging indoor LBS market. However, LBS technologies are not necessarily the primary concern for these big players, enabling new entrants to play vital role in the growth of the industry by specializing in one part of the value-chain. The three strategies for the Startup are: (1) focus on providing niche technologies; (2) partner with big players (i.e., wireless carriers, internet content providers); or (3) challenge big players with superior products.

The purpose of Scenario 3 is to explore a roadmap for a wireless communication infrastructure provider ("telecom") such as T-mobile and to construct two separate business strategies such a company could follow. T-mobile is a cellular communication provider which is now expanding indoors with their Wi-Fi networks. As already mentioned, communication infrastructure providers will have the upper hand in most business deals since they have a lock-in of users using cell phones running off their communication networks. Nevertheless, since the main focus of the communication providers is to improve voice services for their customers, these players will mostly likely outsource their customers' position data to third party LBS application developers. The two strategies are related to the communication infrastructure provider's system architecture, one being a closed-architecture approach, while the other is an open-architecture approach.

In summary, this chapter is a discussion of three scenarios for three different companies operating in various slots within the LBS value-chain:

- Scenario 1: A "Stovepipe" company

    - Strategy 1: End-to-end solution provider approach (niche space)
    - Strategy 2: Incremental growth approach: specialization and outsourcing
- Scenario 2: A "niche market" / specialized (niche market application) company
    - Strategy 1: Niche market approach (niche application…)
    - Strategy 2: Incremental growth approach: partner with big players (i.e., wireless carriers)
    - Strategy 3: Challenge big players (i.e., wireless carriers) approach
- Scenario 3: A "big-player" company (i.e., cellular wireless or non-telecom Wi-Fi provider)
    - Strategy 1: Closed-architecture approach: control location data
    - Strategy 2: Open-architecture approach: outsource location data

The following examines these three scenarios, analyzing the strengths and weakness of the associated strategies. In addition, mistakes made by the outdoor LBS market firms are identified that give some caution to the emerging indoor LBS market on how to approach, for example, partnerships with big players.

7.2.1 Scenario 1 – A "Stovepipe" Company

This scenario considers a Stovepipe company within the indoor LBS value-chain, which is doing vertical implementation across all of the infrastructure types except. For discussion purposes MIT is used as the Stovepipe company, but many of the LPS systems providers reviewed in **Chapter 4/Appendix A** can be considered.

With respect to the communication infrastructure, MIT has its own Wi-Fi network, provided by MIT Information Systems (IS). Regarding the positioning infrastructure, MIT Cricket is the LPS system in itself. Moreover, MIT Facilities is the mapping infrastructure provider, providing CAD floorplans. Also part of the mapping infrastructure is MIT Cricket providing map/content data such as positions of printers or other services (refer back to use case one in **Chapter 2**). MIT Cricket is also the software infrastructure provider, as it developed the WayFinder and ViewFinder applications (**Chapter 2**), as well as the necessary software platform and components. Furthermore, it is possible for MIT IS to also become a positioning infrastructure provider by leveraging their Wi-Fi network and their database of Wi-Fi enabled buildings and rooms. This positioning concept, which we refer to as 'Wi-Fi Map It," is explained in **Chapter 4**. Note that even though MIT is a Stovepipe company, it has several departments.

Moreover, in most cases, the main focus of these Stovepipe companies is the positioning infrastructure, developing location-based technologies and positioning determining methods, which constitute the main objective of the indoor LPS systems reviewed in **Chapter 4**. In order to test these LPSs, all of these companies prototyped indoor LBS applications. As the indoor LBS market matures, however, we believe that these companies will specialize in one or two specific infrastructure types (no more than 3 slots in the detailed LBS value-chain). With respect to MIT and its unique LPS system, MIT Cricket, we believe that it will specialize as a positioning infrastructure provider. With respect to Microsoft, which is a software company, we believe that it will be a software infrastructure provider, developing applications as well as the software platforms. This belief is further explored in the second business strategy.

Finally, it is important to realize that these Stovepipe companies are monopolies in their particular indoor settings (space or community). MIT is operating within its own academic campus. Note, however, that other types of niche spaces or communities are possible, which include airports, convention centers, and shopping malls, to name a few. Also note that wireless carriers are not considered in the indoor settings that these Stovepipe companies operate in since these companies also function as the communication infrastructure providers. Also, communication networks might already exist in most indoor settings prior to the entry of a Stovepipe company, which will then, in most cases, be leveraged by such a company to enable it's positioning infrastructure (**Chapter 4**). Nonetheless, wireless carriers, nonetheless, will play a significant role in the indoor world due to their cellular wireless coverage and the recent Wi-Fi coverage in both, the outdoor and the indoor world.

Also, cellular wireless carriers will start penetrating the market with their Wi-Fi communication networks for the indoor world. This is already seen with partnerships such as Starbucks or McDonalds that provide Wi-Fi networks from T-mobile, Verizon, etc. Nevertheless, in settings such as university campuses where the communication connection is free, users would be more inclined to use MIT's Wi-Fi network to access the LBS services, instead of paying usage fees to access the Internet from their wireless carrier's network.

Overall, holding several slots in the value-chain provides benefits of holding specialized complementary assets. However, standards will need to come into play at some point, even if one LPS/company holds majority of the market share. Take Microsoft, for example, as their evolution to adopt their software to the Web Services/XML paradigm (**Chapter 6**).

Strategy 1: End-to-End Solution Provider

This strategy explores the basic components needed within each infrastructures type for deploying an indoor LBS application. MIT Cricket's WayFinder application is used for discussion purposes. The purpose is to analyze the choices and tradeoffs a company should consider for their business and technology plan. **Table 7.1** shows the overview of the Stovepipe broken down by the infrastructure types it provides with respect to its different departments.

Scenario Players/Providers Overview	
Setting:	"Niche place" (campus, airport, convention center, airport)
Infrastructure type:	
1. Communication	
Entity/Provider 1:	MIT IS Wi-Fi network
2. Positioning	
Entity/Provider 1:	MIT Cricket (absolute position data)
Entity/Provider 2:	MIT IS 'Wi-Fi Map It' (relative position data)
3. Mapping	
Entity/Provider 1:	MIT Cricket – users locations & floorplans (raster maps)
Entity/Provider 2:	MIT Facilities – floorplans/maps (raster/vector maps)

	Entity/Provider 3:	MIT IS – user locations (relative) & Wi-Fi enabled rooms (symbolic)
	4. Software (Services)	
	Entity/Provider 1:	MIT Cricket – Floorplan ActiveMap
	Entity/Provider 2:	MIT Facilities's Map Website

Table 7.1 Stovepipe Scenario Overview

The following discussion explores which issues are more important, what are key choices, tradeoffs, and decisions that impact the timing and practicality of integration and interoperability of the various components in each of the infrastructure types, and what buildup or incremental steps are likely, desirable, or useful.

Communication Infrastructure

When thinking of the choices for communication/networking, it should be realized that in many environments of interest like shopping malls, schools, convention centers, hospital, etc. the communication infrastructure already exists that can provide data networking capability to mobile hosts. After the deployment of the other infrastructure types that are essential for indoor LBS services, such services will complement this useful data networking capability of, for instance, RF wireless LANs. This in turn, will add value to such a network. This makes a wireless LAN more valuable and can increase the chances of its large-scale deployment.

As discussed in **Chapter 4**, many indoor settings such as convention centers, shopping malls, and hospitals already have a communication network deployed. Hence, Stovepipe companies like MIT or Microsoft will probably leverage the existing communication infrastructure in order to decrease deployment costs. In terms of "seamless" communication infrastructure, roaming between APs is supported, and Wi-Fi networks, for example, can be extended to create "clouds of connectivity" inside the so-called hotspot (i.e. locations with high connection frequency such as an office building).

Positioning Infrastructure

Deploying a LPS system that works well indoors is a challenge, because signals reflected off walls, floors, and ceilings tend to confuse sensors. **Chapter 4** (**Appendix A.1/A.2**) discussed the choices for sensor technology type (RF-based, IR, ultrasound, etc.) and positioning (geo-location) methods (TOA, fingerprinting, etc).

It is preferable to employ and leverage the existing communications infrastructure (i.e., Wi-Fi networks) to determine the position of users. This will decrease the deployment costs of indoor LBS services. Exploiting this existing and growing communication infrastructure can provide indoor positioning comparable to what GPS is to the outdoor world. Different indoor LBS applications have different requirements for accuracy and timing. As discussed in **Chapter 4**, determining location based on using a Wi-Fi network does not provide the best positioning accuracy, but, nevertheless, users could be positioned to room scale accuracy (with additional measures such as triangulation and/or adding another sensor type like ultrasound to compensate for the low precision of Wi-Fi based positioning), which is adequate for many indoor LBS applications.

MIT Cricket could potentially be set up through the campus (indoors) to position users in every room/space. An economical question is wheatear the company's LPS system is based off the communication network. In MIT's case it does use the existing RF communication network installed in its building (and could do the same for a wider deployment across the campus). However, MIT Cricket requires extra investment in the positioning infrastructure that consists of ultrasound beckons, which makes the positioning infrastructure investment more costly (i.e., installing, configuring beacons) (**Appendix A**).

However, the tradeoff for higher cost is the higher positioning accuracy that comes with MIT Cricket. For many indoor LBS applications, accuracy requirements are more precise than one might think. For example, for most applications it is essential to be able to tell with certainty which room a person is in, or which side of a partition they are on. For this reason, accuracy to within a foot or so is required for many of the indoor LBS applications.

In most cases a simpler approach (i.e., a single tag and sensor per room like Active Badge instead of multiple sensors per room like MIT Cricket) might suffice and be significantly cheaper for some basic indoor LBS application. However, it is important to think of the positioning infrastructure long-term wise as there are additional LBS applications that might become interest that can only be implemented with a more precise LPS. Hence, it is important to consider whether the simpler approach will suffice for all indoor LBS applications that may be needed in the long term. If not, it makes sense (due to the cost of any positioning infrastructure) to implement a positioning infrastructure that will handle all LBS applications from the simplest to the more sophisticated.

Possible alternative is using ad-hoc networks, which rely on mobile-devices (users) for the exchange of location information. A system like MIT Cricket could use a sparser ceiling-mounted ultrasound receiver grid if the listeners (mobile devices) could also accurately measure their distance from other listeners and share this information with the infrastructure (service-based). See **Appendix B** for more on network-based and ad-hoc networks. Another possible alternative is to use of an alternative positioning sensing technology like UWB and/or a different LPS like MS RADAR.

Overall, positioning and location is a hard problem, especially in buildings where multipath signals tend to ruin the position accuracy. UWB achieves higher accuracy due to its wide bandwidth solving the problem of multipath in an indoor environment. Also, compared with radio frequency identification (RFID), UWB has better range, which makes it better for positioning/tracking. RFID tags generally must be within six to eight feet from scanners, while UWB tags (e.g., Ubisense's UbiTags, **Chapter 4**), for example, has demonstrated its UWB system at over 150 feet (though with reduced accuracy).

Mapping Infrastructure

There are various location model types – symbolic, geometric, and hybrid – that are discussed in **Chapter 5**. It's important to realize that different LPS systems express location in different ways – different measurements (geometric versus symbolic), different spatial frame of reference, or different uncertainty. As a result, it is important to first determine which type of positioning (absolute versus relative) the LBS service will require. In **Chapter 8**, we present filters that help a service designer determine the necessary steps moving along the infrastructure types.

Software (Service) Infrastructure

MIT Facilities has developed a mapping website. The website can map not only the building locations, but also the rooms and floors that have Wi-Fi coverage. The software architecture is set up in a way that users could enter their XY coordinates and be displayed on the map in a similar fashion as with the Wi-Fi coverage mapping. Ideally, users would want the LBS service to locate them automatically without having to enter any input (this is a "location-aware" type of a LBS service explained in **Chapter 6**). The mapping website could be utilized on mobile devices and made into a LBS service using MIT Cricket for automatic location positioning. However, MIT Cricket would need to be adopted, if it isn't already, to locate users according to the floor. Users would then be able to request such a LBS service to locate them and map the nearest bathroom location, for example. This service could also navigate the user to the bathroom of his/her choice, which would require either a route map result or text directions result (refer back to the navigation use case in **Chapter 2**).

MIT Cricket also has its own mapping software application, Floorplan ActiveMap (**Chapter 2**) for a mobile device. As the user moves in a building, the navigation software running on the mobile device uses the listener API to update its current position. Then, by sending this information securely to a map server, it can obtain updates to the map displayed to the user.

In addition, mobile devices learn about services in their vicinity via the Floorplan application that is sent from a map server application, and interact with services by constructing queries for services at a required location. Services appear as icons on the map that are a function of the user's current location. The services themselves learn their location information using their own listener devices, avoiding the need for any per-node configuration (see service discovery in **Chapter 6**).

Use of Open Interoperability Standards

The relationship and interests between the three players and why the need for possible coordination and deployment of standards is as follows. MIT Facilities has a campus wide focus and current facilities management needs so they have no current need for MIT Cricket scale location/activities. MIT Cricket, being currently a just 'in the lab' project, doesn't need to expand their deployment beyond their building to experiment with their LBS ideas. Similarly to MIT Facilities, MIT IS also has a campus wide focus and has no real motive (or business incentive) to deploy standards in order to ease coordination with the other two players.

Trying to do too much coordination too soon will slow down either project (i.e., using cumbersome database management packages, handling many users or too much detail, worrying about editing and updating, trying to keep all the evolving interface standards in sink, etc).

However, MIT Facilities recognizes that, for example, a common coordinate system is essential for doing facility management, especially in an environment such as MIT's where one building is actually a series of 14 separate interconnected CAD plans that don't necessarily know where they lie on a local/global spatial reference system. For this reason, MIT Facilities invested in figuring out what the correct orientation is for each floor plan (i.e., a world file) to allow the construction of a room location model based on real-world coordinates. This is very important because MIT Facilities can now do horizontal and vertical adjacency studies not allowed with the CAD files as they were and MIT Facilities can now overlay seemingly dissimilar data sets and perform analyses not possible before. A simple example for this would be to know where the closest fire hydrant is to a particular lab with a particular use. This is querying architectural information with something that is typically a civil/survey question. Even though this is not directly related to indoor LBS applications, MIT Facilities is aware that it does open the floodgates to such applications like indoor location services (i.e., way finding, emergency management, etc.)

Also, MIT Facilities recognizes that in the future they would like to keep track of equipment in rooms and capabilities at a higher scale (higher granularity). This means that a positioning infrastructure will need to be deployed in every room, which would pose an incentive for MIT Cricket to expand. If MIT Cricket wants their position data of the users to be overlaid on top of the GIS maps from MIT Facilities, it needs to be in the same data format and same coordinate reference system. Use of GML for data format (for data display and integration) and LIF MLP API would take care of the same coordinate reference system.

Also, both MIT Facilities and MIT IS could benefit from a common data model; The development of detailed models is a costly task (especially when extending them to sub-detailed world models). Therefore, different applications should be able to share the same model information. Having such a common model may increase interoperability between applications and make new classes of applications possible. The basic requirement for such an approach is a common language for describing and querying location information.

Similarly, MIT Cricket knows that their LPS must eventually be possible to integrate on top of the existing Wi-Fi networking (communication infrastructure) and Web service (software infrastructure) to avoid Stovepipe implementation (**Chapter 6**).

Overall, it is difficult to exchange information between all these players as their systems are standalone and vertically integrated. Data conversion would be required for integration. Nor is it possible to notify the applications on one system about information changes based on the information sensed by another system. In order to make the exchange of information more interoperable, open standards from LIF for positioning (**Chapter 4**) and OpenGIS for mapping (**Chapter 5**) as well as OpenGIS/OpenLS for software/services (**Chapter 6**) could become of use. As a consequence, all players have an interest in open interoperability standards that would ease integration and facilitate coherence and long-term goals while being easy to do and inexpensive (in time and money and performance) in the short run.

If all departments would follow OGC's WMS specification (**Chapter 6**), for example, Web based maps from both would overlay without any need for conversations and transformations. Also, if MIT Cricket would use LIF's MLP API for positioning, then both the communication handover and positioning handover would be seamless (**Chapter 4**) if users would, for example, travel from the outdoor world where the GSM cellular network is present, to the indoor world (where MIT Cricket's or MIT IS's Wi-Fi network is present). In short, this MIT Facilities and MIT Cricket is a good example of how individual systems/projects start and developed over time.

Strategy 2: Specialize & Outsource

Usually infrastructure providers will be reluctant to commit significant internal resources to deploy all type of infrastructures and to build their own LBS application solutions from the ground up. Technical issues, market understanding, and legal and financial LBS value-chain complexities will continue to encourage these players to rely on other providers (**Table 7.2**) with experience and existing links to the content and applications community. These are factors that we consider in **Chapter 8** as part of our dimensions and filters for consideration when designing LBS services.

These issues are particularly important, given the experimentation and flexibility that happens during the initial stages of LBS services rollout. The service/content providers will be concerned with the success of the new services, as well as the assignment of benefits and revenues generated. For some players, there will be direct cash flows, while for others, LBS services may translate into branding opportunities that will generate profit further down the transaction line, or service stickiness that reduces churn and attracts more premium customers. Again, we discuss these issues and uncertainties in more depth in **Chapter 8**.

Communication Infrastructure Provider	Positioning Infrastructure Provider	Mapping Infrastructure Provider		Software (Service) Infrastructure Provider		
		Base Data & Models	Content & Aggregation	Platform	Application Development	Service Integrators
- Wireless carriers (T-mobile) - local Wi-Fi networks (MIT) - regional Wi-Fi networks (NYCWireless)	- MIT Cricket - MS RADAR - HP Locus - PinPoint - UbiSense - TruePositions	- MIT Cricket - MS RADAR	- priceline.org - ebay.com - Mobile 411 - MapQuest.com - Vindigo	- MIT Cricket - MS RADAR - HP Locus - Oracle - ESRI - Webraska	- MIT Cricket - MS RADAR - HP Locus - PinPoint - Yahoo Find-a-Friend	- MobiSPOT

Table 7.2: Detailed LBS Value-Chain

A real-world example of a LBS services aggregator is MobiSPOT[102]. Being the first in its kind, it sets the standard on how to design and market a professional LBS service. The MobiSPOT aggregator platform of Teydo enables third parties to enhance their applications and services with positioning, messaging and billing functionality.

As we described in **Chapter 6**, most LBS applications require good integration with other components in the system. LPS technology has some demanding real-time requirements, so systems tend to use distributed processing and in-memory databases for low-level tracking functionality. This is another factor in favor of established specialized systems like Oracle. Any company trying to establish itself in this area faces a difficult challenge in achieving these database functionalities since Oracle is already the established provider (among few others) in this niche.

Moreover, integrating the different functions that are required in an application server is not an easy task and requires specialized development skills for each of these areas, which resulted in the commercialization of these products to ease the development of LBS services, especially in its beginning phase. Webrakska, for example, functions as an application service provider (ASP) making functions (i.e., positioning, geocoding, reverse-geocoding, mapping, routing, spatial searching (ranking) available to programmers through an API on its own server. Similarly, ESRI's ArcLocation Solution is a middleware for LBS services that operates between the application servers (i.e., GIS) and the communication/positioning network. Both Webraska and ESRI, like many others in the LBS industry, use and provide standardized APIs such as the OpenLS APIs discussed in **Chapter 6**. In the absence of a standard API provided by the various vendors and to the various services, integration or the bundling of such services would take extra time, effort, and requires specialized knowledge. Note that both vendors are from the outdoor world, but could adopt for the indoor world.

Hence, specializing in one of the infrastructure types is a good business approach for the Stovepipe. Specializing as a platform provider, for example, we speculate that appropriability will loosen as others learn how to build competing platforms. First movers in the platforms arena such as MIT Cricket (or MS RADAR) can offer toolkits that allow third party application developers to develop software that fully integrates into their platform using open and standardized APIs. This should provide high switching-costs and, as a result, lock-in of existing customers.

Also, this strategy addresses the question, "*who actually sells the service?*" Will the end-user choose to order a LBS service through the operator owned portal, or will other channels win? The issue is highlighted by the dramatic development of the ring-tones and logos market. With focused marketing from non-operators, the market suddenly ballooned, and it is estimated that nine out of ten ring-tones are sold through non-operator portal channels.

---

[102] MobiSpot: http://www.mobispot.net

7.2.2 Scenario 2 – A Specialized (Niche Market Application) Company ("Startup")

This scenario plays out the roadmap for a company, referred to as Startup, which specializes in developing the Buddy Finder and the Product Finder (**Chapter 2**). Indoor LBS applications cross a variety of domains (**Table 7.3**). The challenge from a business and application design standpoint is to narrow and specialize the functionality for the particular target market. For example, the focus of the Product Finder is on knowing the static content information, which is quite different from the Buddy Finder focused on the mobile/dynamic whereabouts of friends, events, etc. LBS applications also cover nearly every attractive wireless demographic market, including: parents, teenagers, singles, college students, online communities, business executives, and entrepreneurs. These individual markets have repeatedly demonstrated the willingness to pay premium prices for services that satisfy their needs and desires.

Types of Locations	Location Services			
	Consumer	Business	Government	Military
Positions (absolute and/or relative)	Where am I? (map, address, place); Where is? (person, object, business, place); "Show me the nearest... (business, place)"	Contact nearest field personal (airport); Status of utility field devices; parking; "What's near this store?"	Location sensitive reporting;	In-building training
Events	"Buddy Finder"; mobile concierge; mobile commerce ("Product Finder")	Job fair promotions; Office Presence	Local public announcements	
Navigation / Routes	How do I get there? (address, place); fastest/best route? (i.e., campus navigation)		Rescue operations	Rescue operations
Directories	Looking for the nearest...(specialist); Where can I buy...? (product, service) ATM finder, restaurant finder, etc.)		Public services	
Security/Tracking	Special Zone Tracking (Wi-Fi/Location)	Hospitals: tracking doctors, patients, and medical resources		

Table 7.3: Types of Location-Based Services

Even though the cumulative total of individual niche markets is seen as more significant to, for instance, the communication infrastructure providers, the successful business model for indoor LBS will depend upon the specific target (niche) market and applications involved, not a one-size-fits-all approach. The business model discussion becomes more complex as more players appear between the original application provider and the end-user. The list of potential players continues to expand. Besides carriers, content providers, third party application developers, and distributors make the list grow to include businesses seeking brand placement. And, in the case of Wi-Fi, players such as national/regional/local hotspot network providers, real estate owners and tenants, and even municipal governments.

Moreover, while there may be wireless applications that use location-based technology for add-on services (i.e., ebay.com, priceline.com), there are several services that will be built entirely around this technology. We believe that the drivers of popularity for this technology will be primarily in stand-alone (niche market) LBS services.

In looking at one of the initial LBS applications, E911 service, there are small network externalities associated with it. The service is mandated by the FCC to help protect cell phone users in the event of an emergency. Users of cellular phones are paying for the service through a tax, and will use the service any time they call 911. This service does not directly become more valuable as more users use it. Therefore, the network externalities are small. However, there are network externalities in the LBS industry stemming from the fact that several stand-alone LBS services are more valuable as more users use the service. The Buddy Finder is an example of such an application that benefits from having more users use the service.

The following entry strategies are explored for Startup with respect to a communication infrastructure provider (i.e., wireless cellular carriers or "telecoms" and Wi-Fi network providers).

Strategy 1: Niche Market

Some software infrastructure providers such as LBS application developers may choose to approach market niches that are currently ignored by the communication infrastructure providers. Note that even though wireless carriers do realize that LBS service bring higher AURP (**Chapter 1**), they are focused on improving existing voice services, allowing an opening for smaller, quicker LBS technology firms to develop applications such as Buddy Finder, that serve niche consumer markets.

Choosing the right niche market application might be a challenge on its own. Most of the LBS applications use small pieces of contextual information and none of them are especially compelling. However, the misassumption within the industry about niche market LBS applications is that there appears to be no single killer application. The natural tendency is to forget that several niches added together can constitute a very significant market in total, and in the case of LBS, a very large and profitable total market that has a very high ARPU, with a corresponding "piece-of-the-pie" benefit to all the associated market players.

Popularity can be measured by more users using LBS, users using LBS services more times per day, or users using LBS services more minutes per day. The Buddy Finder application becomes more valuable as more users use it. Hence, locking a user base that is large enough should be the main business goal. Starting with the most simplest form of an LBS service might be essential to make mobile users adopt to the technology. For example, sending a simple map of a shopping mall onto the user's mobile device might be enough to start the adoption process towards more comprehensive LBS applications (i.e., Buddy Finder or Product Finder).

Moreover, the applications that build the largest user base first are most likely to capture a large portion of the market and lock future competitive threats. With respect to the Buddy Finder application, to build the largest user base fastest, the provider (and its partner technology providers in the software slot of the value-chain) could offer their service at a discount initially, and later increase prices once the customer is locked-in. This is especially true if the Buddy Finder provider develops a closed standard such that competing similar applications are not compatible.

This is a good example of a company taking advantage of a closed, proprietary standard to exploit the benefits of network externalities. If successful, it will allow the company (and its partners) to completely extract dominant rents. In **Chapter 8**, we explore the factors that go into a decision that an IT manager must make regarding the adoption of standards for IT development, specifically about the particular standards that relate to current development projects of such application like the Buddy Finder.

Other services such as Mobile 411 applications, which enable users to look up points of interest in areas local to them, do not necessarily have a direct advantage to having more users. However, within the whole industry, there is a secondary effect when more users use the service. The effect takes place because when there is a bigger user base, more companies will want to introduce applications, leading to a better service for the customer. However, this more fulfilling service will only happen if the stand-alone applications can integrate. Open interoperability standards are essential to achieving this integration.

In terms of revenue, the communication infrastructure provider could consider charging companies (the places of interest) for the number of times information about them has been searched via the Mobile 411 service, essentially for click troughs. Unfortunately, this revenue model has not been very profitable on the Internet and may prove the same for LBS services.

From a technical point of view, choosing the Product Finder over the Buddy Finder application might be harder and more costly to deploy since a more accurate positioning infrastructure is needed (see **Chapter 4**).

Also, with the advent of Web services (**Chapter 6**), service-bundling has the potential to provide huge market entry opportunities for new players. New opportunities may be available for some service providers to target niche markets in the cases when the backend services are expensive, when service chaining requires specific domain expertise, or when the data provided is sensitive to local context and mobile subscribers. Nonetheless, these opportunities will be limited by the availability of data/service repositories and catalogs in the market. Such players are likely to wait for enough services to become available on the market, and select partners from the players that provide them

Overall, the strength of this strategy is that direct competition with the communication infrastructure provider is avoided. One weakness, on the other hand, is that application developers may not be able to gain access to niche markets without first partnering with a wireless cellular carrier or Wi-Fi network provider due to access restrictions to customer location data. This issue is further explored in Scenario 3, which discusses two strategies for communication infrastructure providers with respect to their system/software-architecture; one is a closed-architecture based on proprietary standards, and the other is an open-architecture based on open interoperability standards allowing third party developers to plug-and-play their components on top of the communication infrastructure.

And, in the outdoor LBS market, firms have made the mistake of originally targeting niche markets but not signaling that they will remain small, ultimately drawing attention of big players like wireless carriers. This situation should be avoided by the startup companies operating in the indoor world.

Strategy 2: Incremental Growth / Partnerships

An alternative strategy for the Startup is to partner with the "big player" (i.e., wireless carriers, Internet carriers) in order to expand into new indoor settings (new markets). For discussion purposes, the Product Finder application (**Chapter 2**) fits better here as it can be compared to a real-world example (from the outdoor world) of MapQuest's agreement with Sprint to provide Mobile 411 services to Sprint's wireless customers. Both MapQuest and the Product Finder display points of interests (address locations for MapQuest; products and stores for Product Finder) and give navigation directions. The strategy allows MapQuest to instantly plug into a large existing customer base of the Mobile 411 service without building out extensive distribution channels. Similarly, the Product Finder application can plug into the Mobile 411 user base and make the overall service more valuable.

Also, companies providing services such as Mobile 411 applications that do not provide end users with increasing value as the user base becomes larger, will have to depend more on developing great user products to successfully compete and capture market share. In this case, the Mobile 411 company that offers fastest and more efficient search capabilities, and more relevant (content) database will win in the end, since end users will base their purchase decisions on the basis of the product features, pricing, and the built-in value that the product brings them.

Unless a company develops a Mobile 411 application with highly differentiated features or pricing, we are more likely to see a high level of fragmentation in the industry, and prices will be harder to extract than in the Buddy Finder application space. One option for companies competing in the fragmented Mobile 411 application space is to form a partnership with a company that can extract dominant rents, such as the Startup company, and to completely integrate Mobile 411 services with a LBS application such as the Buddy Finder.

Other examples of beneficial partnerships are with companies that provide (or aggregate) Internet content, such as Ebay.com or Priceline.com, which, when location-enabled, would be valuable in a shopping mall setting extending the virtual shopping experience into a location-based shopping experience. Overall, more partnerships will mean a bigger user base, which, in turn, will cause more companies introducing applications, leading to a better service for the customer. However, this more fulfilling service will only happen if the stand-alone applications can integrate. As explained in **Chapter 3**, open interoperability standards are key to achieving this integration. Due to the fact that more players will take part, there will be additional need for system/service integration. This, in turn, will increase the need for open interoperability standards that will ease this integration.

This strategy answer the questions, "*how do bundled services affect competitiveness in the market?*" Combining applications such as the Product Finder (developed by the Startup) with Mobile 411 (offered by the wireless carrier) into one, more comprehensive service, does not only make this service more attractive to new users/customers, but also allows the Startup to tap into a bigger customer base, making the overall bundling more profitable for both companies.

However, this strategy is not without its weaknesses and requires a high degree of appropriability to assure sufficient leverage for negotiating partnership agreements with the big player. Some firms in the outdoor LBS market have made the mistake of signing an exclusive parenting agreement with a single wireless carrier, and, hence, limiting overall access to the larger markets.

Also, the Startup might have to adjust to market pressures or trends. The company Apama that we mentioned in **Chapter 6**, for example, has developed an innovative approach for indexing queries. When Apama started, it was initially looking at the LBS market, but the company changed its focus to real-time financial applications when the market didn't develop as hoped. However, Apama's technology is still applicable in spatial applications, and the company is closely watching the market.

Strategy 3: Challenge Big Market Players

As already mentioned, the big players in the LBS industry include the communication infrastructure providers (i.e., wireless carriers). Another strategy for the Startup is to leverage its superior location-based application technology and compete directly with the wireless carriers, allowing for huge potential returns on technology investments should their strategy be executed successfully.

The Buddy Finder application is a good example to position a company's business plan with this challenge approach. This approach is proven to work with an outdoor world example of the startup ImaHima. This startup has developed a strong user base that places great value on their location-based instant messaging capabilities (including interoperability with the popular desktop IM like MSN). New users seeking this technology often manually entered ImaHima's WAP URL soon forming a large user base even before ImaHima was an official parent of any wireless provider. Soon after, ImaHima was able to leverage its large user base to negotiate very favorable partnership agreements with multiple wireless providers (i.e., NTT DoCoMo, KDDI, JPhone-Vodaphone, Swisscom, Orange, Sunrise). Other partners include AOL, ICQ, SwissTXT (content providers that enable 'community building'), and Sulake, Habbo Ltd., Openwave (technology providers).

Also, if sufficiently interesting information can be provided by the Startup, and when its brand is strong enough, the Startup might consider asking the operator to pay for generating traffic. While that is unfamiliar thought to many operators (who believe they have unbeatable brands and own the customer) with the emergence of "virtual operators" (who do not own networks) and the current economic squeeze decreasing traffic volumes, operators might consider different models for increasing their traffic as well as increasing their income. This is especially true in Europe, where operators have paid a lot for 3G networks.

In contrast, some niche markets are destined to be consumed as they were not niches as much as ideas within existing markets that the main strategic players had not thought of yet. We are not saying that many single-purpose products are not worth while; those could especially be grown by vertical markets through their software and product engineering cycles.

7.2.3 Scenario 3 – Big Players ("Telecom")

The significant player in the indoor world will be the telecommunication carriers/operators, just as is with the outdoor LBS services because of the market penetration with mobile cell phones. However, there is uncertainty how Wi-Fi networks will impact Telecoms.

Strategy 1: Closed-Architecture Approach

Throughout the formative years of LBS (1997–2001), the outdoor world's communication infrastructure provider's (i.e., wireless carriers) approach to LBS application deployment was to implement a core node within the Signaling System 7 (SS7)[103] network that was capable of extracting the location of mobile devices from the various components of the communication network.

Two significant components of the communication network were standardized, which are the mobile positioning center (MPC) in IS-41 networks and the gateway mobile location center (GMLC) in GSM networks. The MPC and GMLC are the coordinating entity in the wireless cellular network system. Standardizing them enabled to hide the backend implementation of telephone system, which include the HLR and VLR databases that store user's position information, and the positioning method (i.e., TOA) from the application developers. This means the application developer only has to know the position value and its quality received from the MPC.

Moreover, MPC and GMLC provide a common access point to network entities and external applications for mobile device location information. This makes it quicker, easier and less expensive for communication infrastructure providers to deploy LBS and related (i.e., billing) services from third party vendors. It is also easier for the external service providers since their applications only need to interact with the MPC or GMLC gateway.

Despite the benefits of the MPC and GMLC, communication infrastructure providers assumed that it would be the responsibility of LBS application developers to acquire their own location data. Unfortunately, there was no standard interface to the MPC/GMLC, and, as a result, each mobile device manufacturer had its own interface. Ericsson, for example, had developed a protocol called the Mobile Positioning Protocol (MPP), which uses XML documents transported over HTTP to transmit the request and response data.

---

[103] SS7 is a protocol developed for the cellular applications.

This model (Phase I), shown in **Figure 7.4**, resulted in integration and interoperability problems with existing systems. Each time a communication infrastructure provider wanted to offer a new application to subscribers on behalf of the application developer, it had to custom implement and integrate the LBS application. Moreover, each application typically used different third party software (i.e., GIS) with its own pre-standard APIs. Therefore, each new application did not easily integrate with existing communication infrastructure provider's systems. Overall, the integration was too costly.

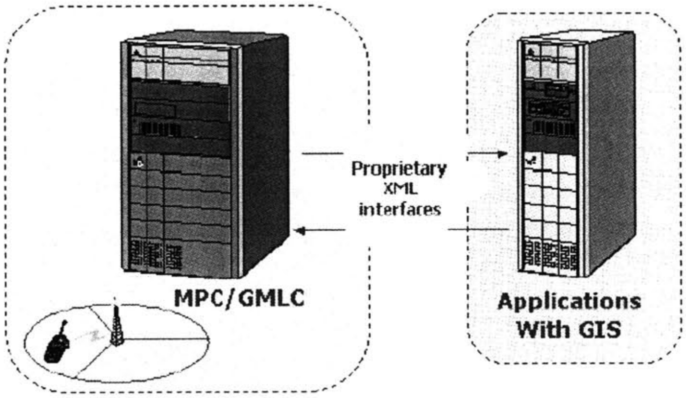


Figure 7.4: Phase I Approach for Communication Infrastructure Provider LBS Architecture (1997–2001 period) (Image Source: ESRI)

In short, the Phase I model simply did not work beyond closed applications (i.e., E911) that served a niche market (singular functions/standalone application), and the industry was forced to define a new model in order to expand communication infrastructure provider-driven LBS offerings. This new model, the Phase II model, is discussed next as the alternative strategy for a communication infrastructure provider. Refer back to **Chapter 2** on the LBS value-chain, specifically niche markets and functionality packaging.

Strategy 2: Open-Architecture Approach

In the Phase II model (2000-2001), shown in **Figure 7.5**, the LBS industry experienced a reactionary shift in how communication infrastructure providers implemented their systems. Several wireless carriers, for example, who had experienced the integration difficulties of introducing standalone applications, recognized the need for a more open-architecture approach. The main benefit of this approach is that it speeds up the process of introducing new LBS applications. Fortunately, this approach did not conflict with the telecom's desire to have system over which they could have complete control, specifically to govern how applications interfaced with their network.

This control of the user's position/location information is an important issue that enhances the need for standards and modularization of elements such as access control/authentication (possible with the LIF MLP API) and storage once the market moves toward many cooperating partners in order to grow and be sufficiently seamless.

As discussed in **Chapter 6**, an open-architecture can be achieved on the principles of Web services, which allows application developers to build LBS applications by using a suite of enabling technologies and APIs.

These applications, which usually reside outside the communication infrastructure provider's firewall in the IP domain, can then be publish back to the communication infrastructure provider's network. This shift in the architecture approach made the LBS industry realize the need to standardize two more components to ease the integration, which fundamentally changed the way the majority of the industry suppliers perceived the LBS value-chain. These components are the location-enabling server (middleware) and the GeoServer (**Chapter 6**), which have open APIs that can be embedded in any LBS application.

Nextel, for example, is the only U.S. telecom carrier to offer an open GPS API to J2ME developers. This offering builds on the open source philosophy and it does not require the application developer to strike a special deal with the carrier for location. The GPS API allows developers to turn their device into a GPS receiver, and from there, the developer can do whatever they want with J2ME. In the UK, Orange and mm02 sell location on a per transaction basis to any GIS developer that has permission to obtain a subscribers location. LBS will not meet its self-imposed expectations, unless all carriers make location readily available through lease or wholesale pricing schemes to all software (services) infrastructure providers (i.e., LBS developers).

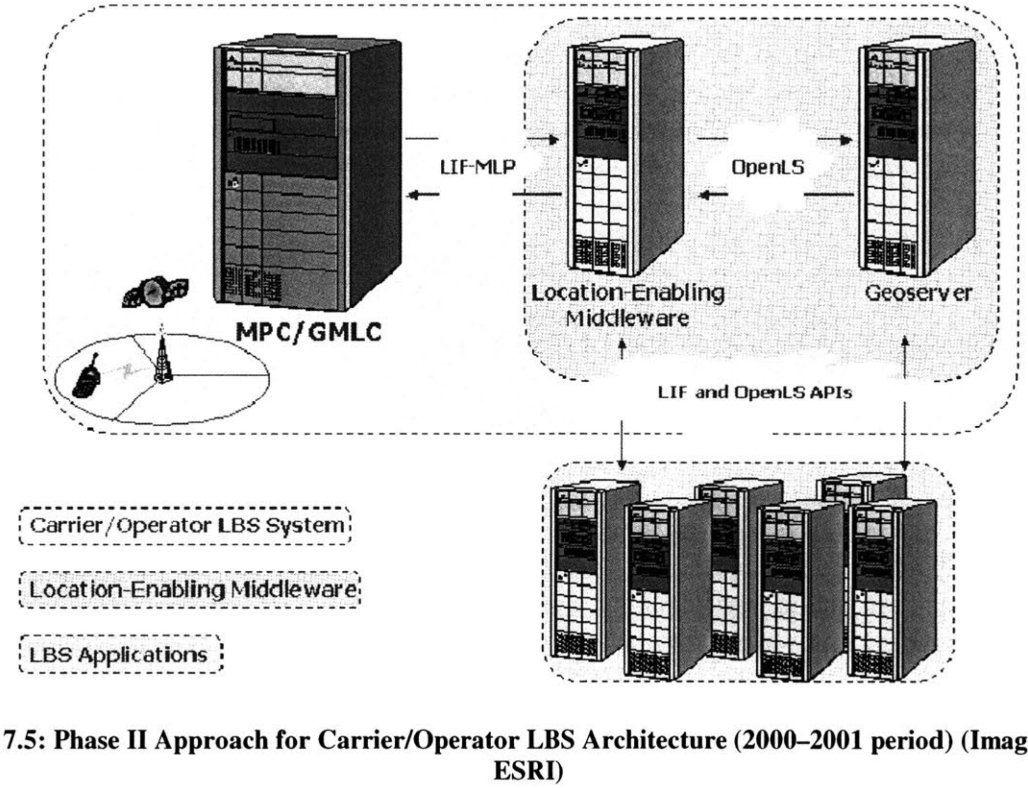


Figure 7.5: Phase II Approach for Carrier/Operator LBS Architecture (2000–2001 period) (Image Source: ESRI)

This model ensures that the communication infrastructure provider has complete control over how any application developer interfaces to the core network, and it dramatically reduces implementation costs. With the phase I LBS architecture, the industry was weighed down with non-standardized proprietary interfaces to closed systems that were neither extensible nor portable. In the new phase II model, however, open interoperability standards are very important. Under the Phase II model, there are two standards that promise to accelerate integration and stimulate LBS application development, which are:

3) the LIF MLP API for getting position / location from a mobile cellular network (independent of the underlying network technology), and
4) the OGC OpenLS API for spatial processing (application, data, and presentation)

The LIF MLP API and OpenLS APIs have solved a majority of the integration challenges that slowed down the outdoor LBS market in its shaping years. Moreover, with the unbundling of systems components, it will not be necessary for the LBS market players to build standalone applications in order to gain a share of the market. The new environment will open the door to small niche players to enter this market with application specific offerings that leverage their understanding of particular industries or processes. New opportunities may be available for some providers to target niche markets in the cases where backend services are expensive, when service-chaining requires specific domain expertise, etc.

These opportunities will be limited by the availability of data/services repositories and service catalogs in the market. Such players are likely to wait for enough services to become available on the market, and select partners from the players that provide them. In the face of the new competition, it is expected that traditional system providers will adopt their business models by offering access to components of their systems through portal-like applications. Overall, the path to realizing the full LBS value market potential requires open interoperability standards for data and metadata exchange in addition to well-defined simple service interfaces (**Chapter 6**).

Also, the communication infrastructure provider, on top of which the positioning infrastructure is typically deployed, usually controls the user location. However, third-party companies usually supply the services (content) databases through syndication. This means that the communication infrastructure provider may not have control over how the data is supplied. As a result, there can be a great variety of data exchange protocols, query methods and data structures involved. The consequent conversion and integration problems are usually significant. This demonstrates the need for open interoperability standards and architectures that will ease the integration havoc.

Similar to the outdoor LBS market, the indoor world of LBS is a collection of services offered by a value-chain of interconnected providers from each of the infrastructure types, such as wireless carriers, software companies, application developers, and content providers. As a result of this multi player environment, we hope that the communication infrastructure providers will eventually deploy the necessary infrastructure, specifically the open-architecture model, for indoor LBS to go mainstream.

In **Chapter 4**, we discussed the challenge of a seamless communication and positioning handover. An open-architecture approach following open interoperability standards such as the LIF MLP API by the communication infrastructure provider is a potential solution to this challenge. In the outdoor world, this change to the open-architecture model was predominantly driven by past failures, successes, and trial and error rather than by proactive innovation and creativity. We hope that the indoor world LBS industry is aware of these failures for its own benefit.

This scenario consists of two communication providers, one for each zone. For the indoor world zone MIT provides the Wi-Fi network. For the outdoor world zone, T-mobile (as the wireless carrier/operator) provides the GSM network. Note, that either of these two providers could co-exist in the indoor or outdoor world. For example, the Wi-Fi network can be extended to the outdoor world. This can be deployed by any of these two providers. The access range from a Wi-Fi access point is about 300m in open space. In this case, Wi-Fi "hotspots" would emerge that would enable communication access from the Wi-Fi network, meaning, it would be free to campus users (instead of having to pay for the T-mobile Wi-Fi network as is seen, for example, with Wi-Fi access at Starbucks). Free Wi-Fi access is popular among public "hotspot' in dense urban areas, like Manhattan, being served by a community groups called NYCWireless[104]. **Figure 7.6** shows the current extent of Wi-Fi hotspots in New York City. These must be public hotspots since there must be tens of thousands of home networks (i.e., Linksys), office networks, etc.

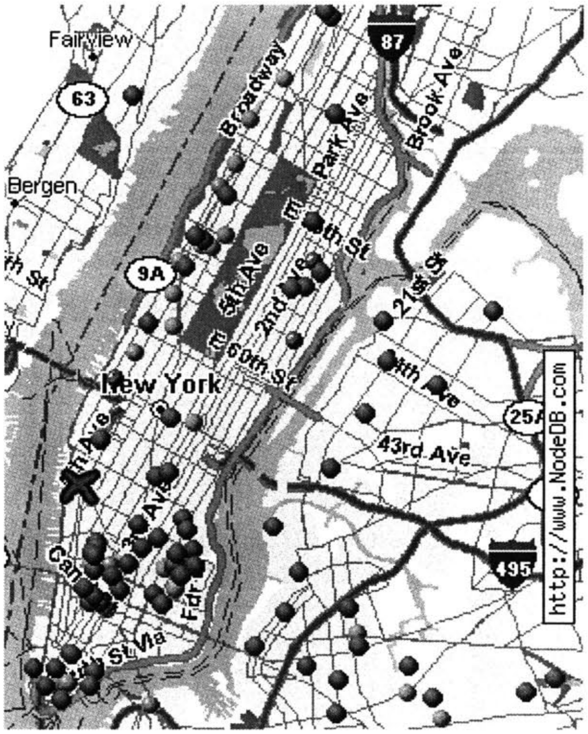


Figure 7.6: Wi-Fi Hotspots in NYC 105

---

[104] NYCWireless: http://nycwireless.org

[105] NYC Wi-Fi Hotspots map, Website, 2003: http://www.nodedb.com/unitedstates/ny/newyork/

7.3 Conclusions

As mentioned in **Chapter 3**, the process of standardization has been important in creating and growing global markets for computing and communications systems. With this understanding that standards are beneficial in terms of achieving the overall potential of the LBS value-chain, there are four types of conclusions to be made based on the scenarios explored in this chapter, which are:

- Identifying the market forces for and against standardizing sooner rather than later
- Identifying places (i.e., APIs) where standardizing should come sooner
- Identifying market developments that will lead to pushing particular types of standards
- Identifying the biggest risks of companies getting far with a proprietary approach

Starting with identifying the market forces that are for standardization is that as the indoor LBS market becomes more mature, there will be an increasing need for collaboration among the different market players. The ultimate benefits will result through interoperability between LPS systems and software from different vendors, allowing for the reuse and exchange of quality data products, with seamless integration of the location information, into any existing software (service) infrastructure that is based on the open-architecture approach.

Also, the fact that standards change the basis of competition can be a market force advocating standardization. No longer is the base technology in question, the real challenge now becomes to focus on what features differentiate the product from its competitors. Also, failure to adopt some standards can mean loss of profit as non-compliance will result in application's incapability to integrate with other applications/system components. This also applies as a risk for companies getting far with a proprietary approach.

Another market force on a smaller magnitude that is for standardization is a leading-edge user community where developers and content producers have a forum to innovate and introduce higher value applications based on open interoperability standards. This user group can break down the cycle of debating whether the infrastructure is needed first or the "killer application."

A market force that is against standardization is the fact that lack of standards creates opportunity for companies to earn profit and market share by branding a specific (proprietary) solution to a problem. Microsoft is the classic example, capturing the market with its Windows operating system, in essence becoming the standard while the industry was still too young to have developed a more democratic (industry consensus-based) solution. Developing a proprietary solution is even to the advantage for small companies such as the "Startup" company explored in scenario 2. With its popular product and first-mover advantages, the company captured a large user base. By using a closed standard, the company prevented future competitive threats as competing similar applications are not compatible with the popular application developed by the Startup. This is a good example of a company taking advantage of a closed, proprietary standard to exploit the benefits of network externalities and a reason for being against standardization.

Another market force that is against standardization is a situation like the one presented in scenario 1 with the Stovepipe company. Following standards would slow down the implementation of individual (standalone) projects/applications. From **Chapter 2** on the value-chain, we know that standalone applications will not make a big impact on the overall marketplace, however, it is necessary to first deploy them before going further with service bundling. Nevertheless, it is important to be aware of the places where standards could fit in and be easily applied without much interference on the speed of the application development process. Hence, we discuss what APIs are worth considering, next.

In terms of identifying the places where standardizing should take place is portrayed in **Figure 7.6**, which breaks down the infrastructure types and applies where LIF's and OGC's standards are applicable.

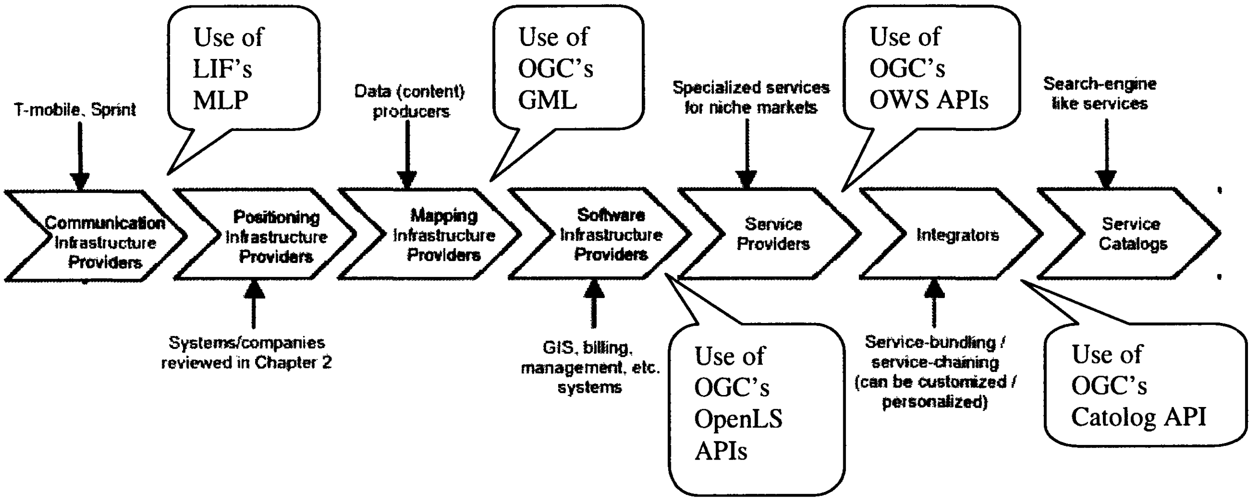


Figure 7.7 Potential Use of Open Interoperability Standard

In terms of identifying the market developments that will lead to pushing particular types of standards, the need for an open-architecture as demonstrated in scenario 3 will push for the LIF MLP API and the OpenLS APIs. From the scenario it is clear that the open-architecture approach enable interoperability, which allows for faster deployment of applications by the original provider or third party vendor. In order to achieve this interoperability, open standards such as the LIF MLP API and the OpenLS APIs are essential.

It is important to realize that most LBS application developers had their own XML (proprietary schemas) during the years of the Phase I model (closed-architecture), which worked fine for most of them who produced modest results and revenues considering the difficulties associated with the decentralized architecture. However, the closed-architecture approach proved to be a disadvantage for telecoms who wanted to implement many LBS applications smoothly and quickly. Also, most GIS vendors were limited to marketing and selling Internet mapping technology to LBS application developers, who in turn built applications atop a GIS that they then attempted to sell to carriers/operators for their commercial offerings.

In terms of identifying the risks of a company getting far with a proprietary approach, it is important to outline and weight on the benefits and disadvantages. Although some may exercise the advantage of being first-movers, the benefit of having tight appropriability on their technologies and applications may not be enough and might lead them into the development of complementary assets and common open interoperability standards to gain market power and extract rent. The exception to this, of course, if Microsoft who is a first-mover, self-contained/stovepipe company, with complementary asset offerings. Nevertheless, Microsoft is following open interoperability standards associated with Web services (**Chapter 6**) since it sees extras revenues and market expansion by enabling its software components to be easily integrated with the ones from other providers. Even though Microsoft is regarded as a monopoly in the software industry, Microsoft is smart enough to know that a one-size-fits all solution is not realistic in today's modularized component-ware marketplace. Actually, even though big players like Microsoft risk losing the profits associated with proprietary solutions, most every major corporation today is involved in the creation of standards.

Also, due to the technical issues, market understanding, and legal and financial LBS value-chain complexities, there is a chance that these pressures will eventually force a company to rely on other providers with experience and existing links to the content and applications community. For example, LBS technologies are not necessarily the primary concern for these big players, enabling new entrants to play vital role in the growth of the industry by specializing in one part of the value-chain. This need for collaboration and specialization, in turn, results in the need for open interoperability standards.

Note, that the hardware infrastructure providers so far have not been addressed in the thesis and their potential impact on the LBS value-chain. In **Chapter 8**, we do consider cell phones manufactures and the driving force associated with hardware.

CHAPTER 8

CONCLUSION: USING FRAMEWORKS, DIMENSIONS, AND FILTERS FOR DESIGNING INDOOR LBS SERVICES

8.1 Introduction

The previous chapters show that the indoor LBS industry is in its "ferment" stage. There is a bottleneck effect slowing down the deployment of indoor LBS applications, such as the ones reviewed in **Chapter 2**. Whilst some amount of this failure can be recognized as a result of the immature infrastructure types explored in this thesis, it has become clear that the visioning of technology and services is not the core problem. Instead, the prototyping and testing of envisioned services, concepts, early standards (**Chapter 3**) and business models (**Chapter 7**) is. Specifically, the concept of having all these standalone applications bundled together needs to be explored further.

We believe that these services cannot be found just by cutting the wires of the existing Internet services. Instead, they must be designed from the ground up to gain added value from the inherent factors of the infrastructure types. With respect to the communication/positioning infrastructures, these factors include location sensitiveness, context dependency, immediacy and mobility, which emphasize the personal roles of the user. With respect to the mapping and software (service) infrastructures, we believe that like within WWW or SMS technologies, the contents (i.e., pictures/MMS (camera phones); games, etc.) that users share with each other may well become the most significant driving force for the adoption of indoor LBS technologies. In other words, the demand for mobile data services will be driven not only by technology, but also by the content of such services, which is already seen in the mobile gaming and video industry.

Mobility also needs to be considered as a factor for the software/service infrastructure with respect to "seamless" service offering and discovery. Service discovery is essential to indoor mobile users, providing the means to exploit services that are offered and to configure end-devices automatically and to register on LBS applications with the minimum (or zero) user intervention. This mobile environment must support a changing collection of services as the user moves within the "multi-world" environment. Considering Web service and the publish-find-bind model (**Chapter 6**) as a technological driving force, providers of the standalone applications from **Chapter 2** would tend to want to have their applications easily discovered and configured to any mobile device.

The consumer of telecommunication services of tomorrow will expect to receive the same services in a wireless fashion as he receives from a fixed network. These services require (at least instantaneously) high bandwidths. It is not expected that future telecommunication users are willing to sacrifice functionality for the added value of mobility nor to pay more for it - mainly because the user will hardly be using any other stationary telecommunication devices.

In the USA, Wi-Fi technologies are being deployed to provide a multitude of high-value services. PDAs of large computer hardware companies have been very successful and combine perfectly with the aspirations of Microsoft to become a world leader also in mobile communications. The E911 regulation for emergency situations dictates that wireless carriers will have to be in the forefront on positioning technologies a fact that is a driving force for LBS services. Question is, how will E911regulations change and impact communication infrastructure providers on the local scale, specifically within the indoor world, such as Wi-Fi providers, both telecom operators and non-operators such as NYCWireless (refer back to **Chapter 6**). Wi-Fi will certainly have an impact on the LBS value-chain. Nevertheless, the E911 initiative is a driving force improving both the communication and positioning infrastructures, on top of which the software (services) infrastructure can be deployed.

Another federal government initiative that could result in being a major driving force is Homeland Security, which can have an impact on the communication/positioning infrastructure due to sensor installment in subway tunnels and other indoor spaces. As a matter of fact, the impact can be as large as the installment of the GPS system by the military. For example, Wi-Fi could be deployed in subway tunnels to allow these sensors to exchange data with the overall sensor network. With permission, the LBS industry could leverage this communication infrastructure, apply Wi-Fi cell-id positioning (**Chapter 4**), and enable LBS services anytime and anywhere. Voice services would also be deployed anywhere, anytime based on this driving force.

Nevertheless, until Wi-Fi is widespread and roaming agreements are in place between them, public Wi-Fi will continue to be more like the Internet cafe experience, where the users have to seek out access, than the cell phone experience, where it finds the user. Companies such as Boingo and Cometa are aiming to achieve this seamless communication handover (**Chapter 4**).

In order to address these issues and uncertainties connected with each infrastructure type and driving forces that may or may not have an impact, this chapter proposes a framework (shown in **Figure 8.1**) that offers a set of viewpoints and steps for generating information to help indoor LBS service designers choose between business strategy and service portfolio options. These steps do not necessarily follow a linear sequence of activities but rather a set of information gathering and processing tasks that can be used flexibly to understand indoor LBS services and mitigate against risks that may accrue from selecting an incorrect design option.

As part of the marketplace dimension, the first step is to find out what the (potential) customers want. In **Chapter 1**, we explored some market surveys done for outdoor LBS services that can be applicable for indoor LBS services. In **Chapter 2**, we explored in detail the use cases for such applications like the Buddy Finder and the Product Finder, which are foreseen to be most promising (with the right business strategy that was explored in **Chapter 7**).

This is followed by two filtering stages that help concentrate the analysis on the most promising applications. The remaining steps deal with the technical, organization, and economic dimensions, and culminate in the selection of a service portfolio and strategy that recognize the high-level of uncertainty surrounding demand.

In developing indoor LBS applications we apply four dimensions (**Figure 8.1**):
(1) Marketplace (what the user wants);
(2) Technical (composed of the four infrastructure types);
(3) Organizational (internal and partners); and
(4) Economic (technology standards)

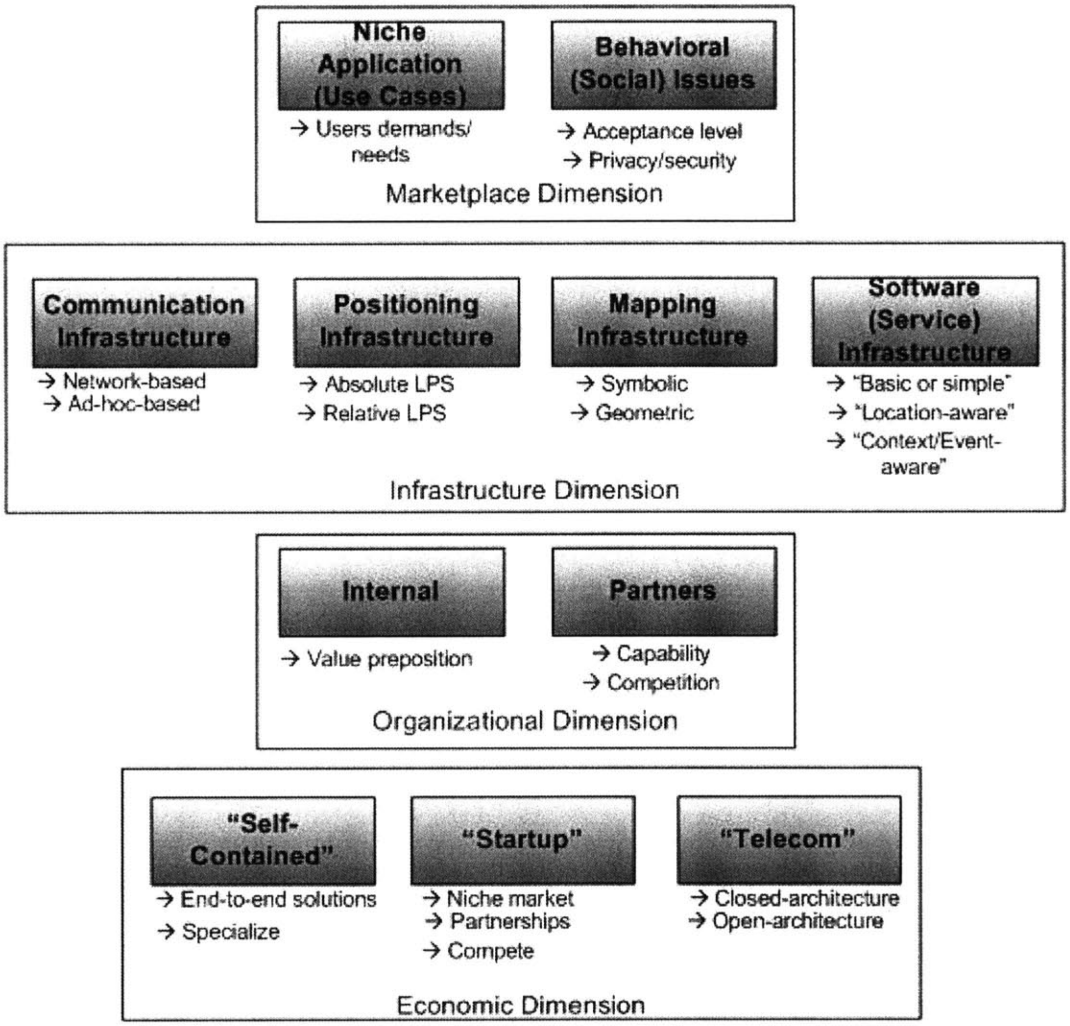


Figure 8.1 Framework Dimensions

8.2 The Dimensions and Uncertainties

The *Marketplace Dimension* captures user characteristics (**Chapter 2**). It also captures the known social and behavioral implications of the service (discussed under the "Constraints" section at the end of this chapter). Services that bring about fundamental behavioral changes in the way people live and work are referred to as social innovations[106]. Uncertainties include:

---

[106] Jarvenpaa, "Internet goes mobile - How will wireless computing affect your firm's Internet strategy," *Working paper*, pp. 1-24, 2000.

requirements definition due to the unknown user context; device diversity and ability to match with technical capabilities; delivering wrong value proposition; demand for mobile services historically difficult to predict (voice, SMS, WAP) -driven by mix of market pull and tech push; little mass market experience with indoor LBS and widely varying estimates of future market size; privacy and security concerns.

The *Technical Dimension* captures the characteristics explored in each of the infrastructure types. For the communication infrastructure (**Chapter 4**) it is about network-based versus ad-hoc based type of service; it includes coverage, bandwidth, cost structure and ownership (e.g. network operator, business, or individual). Consider, for example, an application that will send potential customers a text message with a special discount offer when they are close to a particular store (refer back **Chapter 2** for the use cases). This might not seem much more complex than the previous example, but it requires several orders of magnitude more server processing and bandwidth. For this type of application to occur, the position of all users needs to be updated frequently. This is an active application, or a real-time service, which may need to handle thousands of spatial updates per second.

For the positioning infrastructure (**Chapter 4**) it is about absolute versus relative LPS systems, it includes coverage, accuracy, frequency of update, absolute/relative positioning. The device's location can be determined either by the device itself, or by a communications network. This distinction could impact the ownership and use of the position information.

For the mapping infrastructure (**Chapter 5**) it is about symbolic versus geometric data models, both, for modeling user position location as well as the features and content data of the indoor world.

For the software (service) infrastructure (**Chapter 6**) it is about "simple or basic" versus "location-aware" (versus "context-aware") type of LBS services; it is also about the nature of the LBS service, which can be either location or navigation; it includes the servers, databases, and development platforms and their characteristics (e.g. size/form factor, computing power, display size, battery life) of different device types (PDAs, phones, and RF tags).

Uncertainties in this dimension include: infrastructure is expensive, takes time to deploy and the cost is front loaded; some technologies have unproven capability, reliability and availability; roll-out of positioning capabilities and other infrastructure elements by communication infrastructure providers; integration of communication networks (seamless communication handover); integration of positioning types (seamless positioning (location) handover); integration of mapping types (geometric + symbolic); integration within the software infrastructure (e.g. geographical databases, middleware or legacy systems).

The *Organizational Dimension* captures the internal competences and resources of organization(s) planning to offer LBS. It also captures the capabilities of partners necessary to offer LBS and includes partner strategies (**Chapter 7**). Uncertainties are broken down into two types. For (1) partners: availability of partners with the right capabilities; competition from potential partners; incompatible strategies and visions; ability to reach an acceptable revenue and cost sharing arrangements. For (2) internal: availability of relevant skill sets; ability for organization to communicate value propositions; ability to reach decision makers in target customer segments.

The *Economic Dimension* captures the competitive landscapes, privacy and security concerns, and, most importantly with respect to the focus of this thesis, technology standards. This dimension encompasses or affects all the previous ones LBS and other mobile services also have the potential, over time, to change the circumstance/situation in which they are offered. Uncertainties include: economic growth; privacy/ security regulation; commoditization of positioning capability via government imposed interface standardization; legal liability for service failure; unanticipated social/behavioral impacts; unknown competitive threat from alternative technical solutions/other players; complex value-chain with many players and diverse business models.

These dimensions are not independent as changes on one dimension can lead to changes on the others. For example technological innovations can make new services feasible (technology push) and lifestyle or organizational changes can create new opportunities (market pull). These interactions create a complex environment for designing and managing all mobile services.

This complex environment creates significant risk for LBS service designers, as they don't know what people want or are willing to pay for. Previous mobile voice/data predictions were wildly inaccurate[107]. There is also uncertainty around the ability of service providers to deliver the required technologies. While service providers can choose to play in several places in the LBS value-chain (as shown with the "self-contained" company scenario in **Chapter 7**) their choices are typically constrained by current internal capabilities. To access the additional capabilities required to provide complete LBS offerings, designers run the risk of choosing partners with conflicting strategic interests.

8.3 Steps and Filters

The design of indoor LBS services present a surprisingly complex dynamic environment for service design. This combined with the high-level of uncertainty makes the design strategy formulation for these services a daunting challenge. Hence, this section on the steps and filters is dedicated as a guideline for managers in the basics of what is important to consider when developing indoor LBS services. The last step deals with decision making regarding standards, what they are, why they are important, and what considerations are important when making the decision to adopt a standard.

Note that the paths are probably different, for example, for telecoms and niche players. Also, in either case, the contextual uncertainty and designers' understanding of the dimensions will inevitably change throughout implementation. There is a continued need to reassess the strategy in light of changes, market pressures, and uncertainties. Hence, designers should watch out for network externalities such as the ones explored in **Chapter 7**. Likewise, testbeds and full-scale

---

[107] Economist, "Cutting the cord," in Economist, vol. 353, 1999, pp. 6.

operational experience should be taken into consideration to support planning and execution of subsequent phases.

8.3.1 Step 1: Determining the "Best" Niche Application (Marketplace Dimension)

The first step is to determine who are the potential users and what do they expect from a certain indoor LBS application, such as the Product Finder and Buddy Finder. First hand knowledge of customer needs gained from market surveys, focus groups or service trials can expand the knowledge of social, economic and technical constraints on services (see "Constraints," below).

There is currently a number of practitioner or academic literature that helps in the design of LBS services based on what customers want, or in the definition of a coherent strategy to pursue LBS service opportunities based on infrastructural capability analysis.

Filter A: Filtering by Size of Opportunity

This filter seeks to weed out less promising applications by selecting the largest and easiest opportunities first i.e. the "low-hanging fruit." The filters essentially provide the high-level market and technical feasibility studies suggested by the new product development literature[108]. In the first filter, the applications with the largest revenue potential are selected. The analysis required to estimate revenue potential varies by the nature of the service, but includes a sizing of the target (niche) markets (e.g. industrial sector or consumer demographic), an assessment of the value proposition for these segments, and an analysis of their willingness and ability of the customers to pay.

Network externalities (refer back to **Chapter 7** for examples) should be considered where appropriate. Likely social and behavioral impacts (i.e., privacy) on the service use can help the designer understand the social constraints on the feasible problem, and thereby identify both commercially attractive applications and those which are not likely to succeed. Comparing the Buddy Finder to the Product Finder application, for example, privacy might be the main issue for the Buddy Finder application since its "location-aware" characteristic (**Chapter 6**) makes the control of the user's position data crucial to the service's adoption rate. In contrast, the Product Finder application can be designed as a "simple or basic" (**Chapter 6**) LBS service, not requiring the positioning and tracking of users.

The applications with the highest revenue potential are passed to the next stage for deeper analysis. The rejected applications are not completely cast aside as the service designer should attempt to identify viable applications that can be integrated together that benefit from economies-of-scope (see service-bundling in **Chapter 6**. It makes sense to look at varying service-bundles that target the same niche market and that use the same infrastructure types. For example, as explored in **Chapter 6**, both the Product Finder and the Buddy Finder applications can function based on symbolic location (i.e., "*product Z is in Store ABC*;" "*Bob is in the mall*"),

---

[108] H. Ernst, "Success factors of New Product Development: A review of the empirical literature," in International Journal of Management Reviews, vol. 4: Blackwell Publishing Limited, 2002, pp. 1.

which means that the same positioning infrastructure (a LPS that provides relative positioning as opposed to absolute) can be leveraged.

Another option is a service-bundle that leverages the service provider's strengths in some other way (i.e., provider offers complementary service like Mobile 411 for the end customer, or in the case of a 'self-contained' company, it offers end-to-end solutions for LBS application developer (see Scenario 1, **Chapter 7**).

8.3.2 Step 2: Determining the Infrastructure Types

The second step is to determine whether the service should start of as "simple" (no location positioning by a LPS system) or "location-aware." As explored in **Chapter 6**, determining user's location might be a minor value-add to the core service (e.g. the MapQuest.com service offers directions to restaurants and other information without automatic position determination). In these cases the positioning capability may be so marginal that it could be excluded, making the implementation easier (see the Filter 2, "Filtering by Ease of Implementation").

Requirements determination is concerned with a move in its own solution space (i.e., existing resources such as the technical capabilities and infrastructures of a service provider). For example, if the existing communication provider has a Wi-Fi network, the solution space would entail leveraging this resource to enable a positioning infrastructure that is based Wi-Fi cell-id positioning (either relative or absolute). The anomaly space, or the requirement space, on the other hand, contains all the potential problems that could conceivably be addressed for different stakeholders[109].

In the context of LBS design the large size of the solution space and the considerable uncertainty around its several dimensions makes the discovery of the 'determining an attractive' solution problematic. Overall, the concepts and requirements for application/service can come from a review of competitive offerings and plans, from a review of practitioner and academic literature, or from brainstorming by various groups from within the mobile industry or attractive customer segments. The literature provides some insight into developing m-commerce functional requirements as mappings. Kalakota (2002)[110] emphasizes the importance of developing a deep understanding of the functional requirements of mobile data customers.

Filter B: Filtering by Ease of Implementation

After the application and type of service have been determined, the second filtering stage involves a high level assessment of the infrastructure and organizational dimensions, specifically the technical and organizational capabilities necessary to offer the applications. This corresponds to developing a hypothetical future solution space for each possible applications and thereby determining requirements for leveraging and transitioning from the current solution space. These requirements and related capabilities give an overall indication of the level of implementation

---

[109] M. Bergman, J. L. King, and K. Lyytinen, "Large Scale Requirements Analysis as Heterogenous Engineering," Scandinavian journal of information systems, vol. 14, pp. 37-56, 2002.

[110] R. Kalakota and M. Robinson, M-Business: The Race to Mobility. New York: McGraw-Hill, 2002.

complexity and what service requirements (the ones determined in step 2, above) can be achieved.

LBS functional requirements detail how the current solution space can be transformed into a future solution space. While there will be many possible internal and partnering solutions to providing the necessary capabilities, the filtering decision can be normally based on the simplest option. The non-functional requirements for the placement are defined by the constraints on the future requirements. For example, the LBS service type switches from "basic or simple" to "location-aware" (**Chapter 6**). Or, the "location-aware" service requires a switch over from relative positioning to absolute positioning to enable precise directions. The functional capability for location management can be broken down into a location positioning capability (positioning infrastructure) and location database (software (services) infrastructure) requirements.

Many requirements are dictated by the business context or consumer business model in which the location positioning capability is embedded. To support a wide range of applications the service designer must also consider standardized interfaces to software components (i.e., databases, GIS servers, etc.) explored in **Chapter 6** and build a capability for organizational, and individual customers to instantiate their own location sensitive databases and describe their own business logic.

Also, as explained in **Chapter 6**, LBS services can also generate high volumes of data. Therefore, service designers must carefully consider the type and quality of user interactions to avoid information overload, and to make sure the user gets a seamless service (**Chapter 4**). The key is to understand the context of usage. The level of complexity faced by a service designer in implementing the service depends on internal and partner (i.e., service integrators) capabilities. Gaps in implementation capabilities may include: service portal and personalization capability, availability and interoperability of software components, content/context management, and transaction processing.

In the first pass through the framework the service designer should probably focus on the simplest and easiest applications within a single market as a starting point. In subsequent passes, more challenging services or bundles of smaller services can be chosen for in-depth analysis. As explained in **Chapter 4**, the requirements for location positioning capabilities also vary that may imply very different sets of functional and non-functional mappings. A positioning capability is important for many applications, but may be as simple as "in the room"/"not in the room." In this case, a relative LPS system is sufficient. Symbolic reasoning (context?) aware processing is also often needed to provide useful descriptions of location and for navigation (i.e. "*when you reach the ATM in the middle of the hall, Store ABC will be to your right*"). In **Chapter 5** we explained the requirements for data modeling that apply to this step.

In short, these four steps are illustrated in **Figure 8.xxx**.

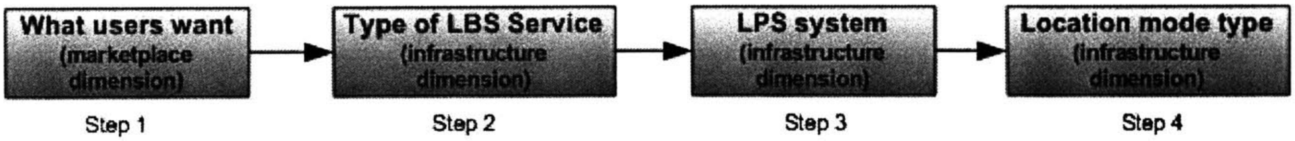


Figure 8.2: General Infrastructure Steps

For example, if step 2 for the Product Finder application determined that the service type is to be "simple," the service designer can draw a conclusion that no LPS system is necessary. However, the user's location can be determined in two ways: (1) user selects his/her location (i.e., shopping mall) based on an aggregation of the place (i.e., shopping mall); (2) user enters his/her location in terms of an address, phone number, etc., for a particular place (i.e., store in a shopping mall).

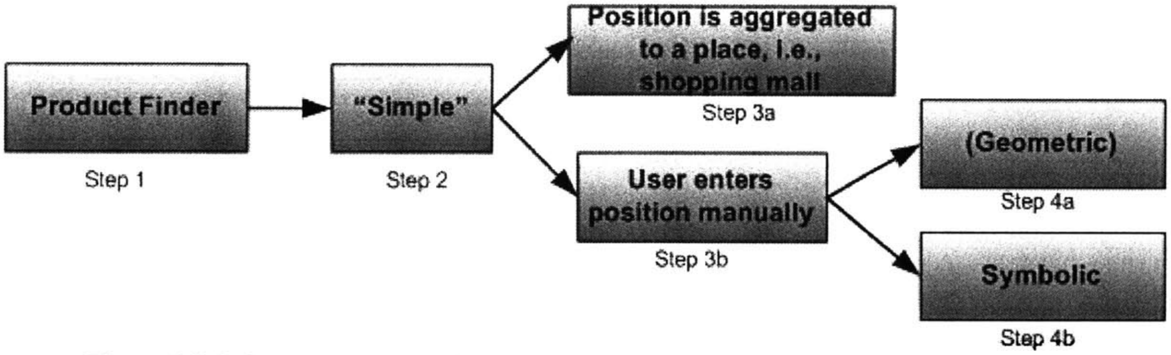


Figure 8.3: Infrastructure Filtering Steps when Designing the "Simple" Product Finder

For a "location-aware" type of a LBS service, the service designer knows that a LPS system is required to position (and track) the mobile user. The next step is to determine if this positioning is to be absolute or relative (see **Chapter 4**). Following, either a geometric or symbolic location data model is needed for storing/modeling user's position (see **Chapter 5**).

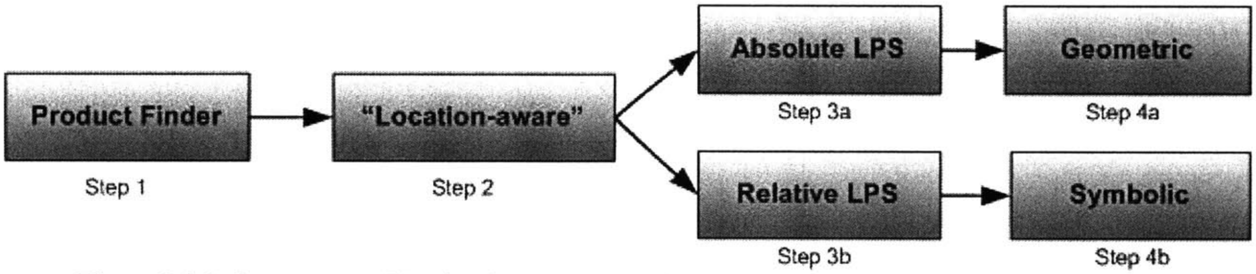


Figure 8.4: Infrastructure Filtering Steps when Designing the "Location-Aware" Product Finder

8.2.3 Step 3: Service Portfolio and Strategy Selection (Organizational Dimension)

Having selected the most promising applications and the service type, as well as using the ease of implementation filter to determine the infrastructure types to deploy, the service designer must decide on the overall LBS strategy associated with those services. It is important to note that this decision is carried out in a background of the uncertainties outlined above for each dimension and due to conflicting strategies and decisions of other players (e.g. partners, competitors and regulators), uncertain market demand, instability of partnerships and switch over of technical feasibility.

Filter C: Filtering by Organizational Driving Forces

The risk and return for potential portfolios is derived for each scenario to gain an understanding of the crucial driving forces behind the attractiveness of services. For each potential portfolio and scenario combination the designer needs to design a strategy that consider the following factors:

- Adoption rate
- Development of product offering stages (i.e., enhancing the "simple or basic" into a "location-aware" service, or service-bundling with external service providers)
- Perspective on the complete LBS value-chain
    - Proportion of value-chain that can be locked-in
    - Commitment to the different types of infrastructure
- Level of commitment to reducing the sources of uncertainty surrounding the marketplace dimension (e.g. testbeds) and infrastructure dimension (e.g. standards).

Analysis of strategic alternatives over a range of scenarios provides insight into the most important sources of uncertainty, and the mitigating actions most likely to increase return and reduce risk. Attractive service portfolios are likely to leverage existing infrastructure investments, internal/partner capabilities or partnership arrangements. Actions to reduce risk include the retention of flexibility where uncertainty is greatest such as by making incremental investments (e.g. cost position, customer/partner relationships) that preserve a "right to play."

8.2.4 Step 4: Assessing the Industry's Potential Impact (Economic Dimension)

Particular attention needs to be paid to the likely paths (affected by driving forces) the industry might take to reach the alternative futures[111]. This allows the designer to identify network externalities that indicate that the industry's roadmap. Scenario analysis must include developing a perspective on where value is captured in the LBS value-chain. Parts of the value-chain that are of critical importance across a range of services are the strategic high ground that "self-contained" companies or communication infrastructure providers should plan to capture (e.g. by specializing and developing an internal capability). In **Chapter 7**, we explored such business scenarios and strategies for these market players.

The output of the strategy selection process includes a portfolio of LBS offerings along with an understanding of the target markets and a high-level plan for the articulation of value propositions. The strategy includes a decision on the service provider's (i.e., wireless carrier's) strategic posture (shaper/adapter), and action plans for creating partnerships, for influencing standards and regulatory bodies, and most importantly, an understanding of the investment for the different types of infrastructures explored in this thesis.

The service provider should plan appropriate options preserving its right to play at a future date and an understanding of the network externalities (see **Chapter 7**) that correspond to resolved uncertainties. The high-level requirements associated with a strategic contextualization offered by the scenarios provide a starting point for the detailed technical design for implementation.

Filter D: Filtering by Technology Standards

---

[111] H. K. Courtney, Jane Viguerie, Patrick, "Strategy Under Uncertainty.," in Harvard Business Review, vol. 75, 1997, pp. 67.

The importance of IT standards has increased over the past ten to twenty years (**Chapter 3**). Increasingly, IT managers must make decisions regarding the adoption of standards for IT development, specifically about the particular standards that relate to current development projects. After reading about this filtering stage, an IS manager should have a solid grounding in the risks, factors, and challenges involved in evaluating the appropriateness of specific standards for specific development projects.

All standards are not created equally. On one end of the spectrum are the highly formalized, carefully crafted standards created by democratic international bodies such as the ISO. On the other end lie market driven *de facto* standards, such as the MS Windows OS, which derive their influence from widespread adoption. In between are standards developed by consortia like the OGC, which seek to maximize the benefits and minimize the detriments/disadvantages associated with both formal and *de facto* standards development processes. It is crucial to find out who is responsible for the creation of a standard, how they have gone about creating it, and what potential traps or benefits exist because of the creation process.

With this basic information the IS manager is ready to begin making the difficult decision about whether or not to adopt a given standard. The manager must look at the criteria for successful standards:

- technical quality,
- timeliness,
- effectiveness,
- widespread adoption

In addition, many other factors must be weighed including:

- the need/desirability for compatibility or interoperability with other systems,
- the existence or cost of acquiring the necessary skills and expertise needed to implement the standard,
- the strategic importance of adoption (product compatibility),
- the stability of the standard, and
- the consequences of not adopting the standard.

In some cases, making this decision will be relatively painless, as in choosing a target operating system or Web browser for a new system. On the other hand, choosing a programming language may pose more of a challenge. The size and influence of a company may also have an impact on the decision.

The following are the main factors to be considered when assessing the pros and cons of using a standard:

Cost-related Issues

Standards are expensive with respect both to financial and human capital. The leading experts in a field must be donated by their employers to the process, and in addition, those employers must put up the cash to transport those people to meetings, organize conferences, etc. As standards in general have become more important the costs, and more importantly the return on investment, of the standards process has come under more intense scrutiny. Issues related to cost are: who pays for the standards development process, how much, what those payments buy them, and what is the cost to users for the resulting standards.

Timeliness

A second major issue that is the timeliness, with respect to market cycles, with which standards are developed. The dominant perception here is consortia standards like the one of LIF and OGC are fast. Timeliness only becomes an issue when one is considering the creation of a new standard. In that case, managers need to make the decision about whether speed is more important than quality. In high risk projects it is crucial to build upon standards of the highest quality since in the event of failure this helps to reduce liability damages.

Legitimacy

For all practical purposes IT managers nowadays are free to adopt standards regardless of their origin. The legitimacy of standards is first tested in the market. At the same time it is very important to understand that all standards are not created equally and now more than ever managers must be careful to research the origins of any given standard, and evaluate the likelihood of market success before deciding to adopt.

Making the Decision

Now that we have a background in the standards process, the following section will provide some realistic guidelines for deciding whether or not to include standards into one's development projects. What are the elements of that decision? There are a number of classes of considerations to be made. The importance of each will vary greatly depending on the particular development project in the particular organization. Those considerations are:

- The quality of the proposed standard (technical quality, timeliness, effectiveness, widespread adoption)
- The type/usage of the standard
- Compatibility/interoperability issues relating to one's current and future IT architecture
- Strategic importance
- Personnel/expertise issues
- Stability of the standard
- Consequences of non-adoption
- Capacity to participate in standards development, or to wage a standards war

We will briefly describe each of these considerations and leave it up to managers to apply them as appropriate to their own organizations. There can be four qualities that describe "good" standards[112].

1) *High technical quality* of the standard is the first element to be examined. Hidden in this requirement is that the IT manager, or someone on the staff, must be competent to evaluate whether or not the specification is of high technical quality or not. Naturally, a standard which is not of high technical quality will result in end products which have flaws or other undesirable qualities.
2) *Timeliness* is the second quality of a "good" standard. Earlier we spent a good deal of discussion on this issue. Generally the strength of the need for a standard dictates the speed with which it will be developed. Recently consortia such as W3C are spending considerable energy on predicting and even directing the future need for standards and planning to develop them in time to meet the demand. The hidden trap here is not to go with standards that exist before their time. Companies may build products to fit a standard whose need is never realized and be left with a warehouse full of gadgets no one wants to use.
3) *Effectiveness* is the third quality of a good standard. Effectiveness is different from technical quality in that a specification may be of high technical quality and still not effectively address the core problem--i.e. it is not an effective solution. Many standards exist which address similar issues or problems. IT managers must carefully choose the standard that is most appropriate to the given design problem. With respect to the LIF and OGC standards, there isn't much (if not zero) overlap with other standards from different standard bodies.
4) Lastly, widespread adoption is perhaps the most crucial success factor for a standard. Certainly this is easy to evaluate for established standards, but IT managers must beware of emerging standards that have not yet reached maturity. With respect to the OpenGIS WMS standard (**Chapter 5**), there are now hundreds of software products that are WMS-compliant.

Using these four criteria IT managers can make a good preliminary evaluation of a potential standard. Provided the standard meets these criteria, the next question is its appropriateness to the development project at hand.

How the standard will be used is very important. In our case, this consideration deals with interoperability and compatibility. There are numerous areas which need to be considered such as backward compatibility with legacy systems, compatibility with the products the user currently owns, and forward compatibility with products a company and others will produce.

Especially in the area of information systems interoperability with very different systems is increasingly an issue. Primarily the issue of compatibility can be seen in terms of switching costs: how much will it cost the company to switch to a new standard, will users of the product

---

[112] Willingmyre, George T. International Standards at the Crossroads. ACM StandardView 5(4), December 1997, p190-194 (http://www.acm.org/pubs/articles/journals/standardview/1997-5-4/p190-willingmyre/p190-willingmyre.pdf)

also be willing to switch, is there a general switching trend going on in the industry, and will the new standard lower switching costs in a way that can help or hurt the company?

The open source movement and standards such as languages built on XML are bringing increasing focus on universal compatibility between radically different devices. The cost of adopting a standard might be prohibitively high for one compatibility reason or another, but in these times the costs need to be weighed carefully against the cost of *not* adopting the standard. There are more than compatibility issues at stake here so these costs will be discussed last.

The other consideration is strategic and include issues previously discussed such as switching costs for users of the products. Applegate et al. (1999) provide a useful set of considerations, asking five questions:

1. Can IT build barriers to entry into a given market?
2. Can IT build (or reduce) switching costs from your product to a competitor's?
3. Can IT change the basis of competition?
4. Can IT change the balance of power in supplier/consumer relations?
5. Can IT generate new products?

By their very nature and purpose, standards:

1) reduce barriers to entry into a given market,
2) reduce the switching costs between competing products,
3) change the balance of power, and
4) generate new product ideas.

For example, standards in the frequencies at which remote control devices operate allow competing vendors to provide remote controls that will work with their competitor's products (e.g. TVs/VCRs), reducing the cost of switching to the consumer. Perhaps question three is the most crucial of all. Standards change the basis of competition. No longer is the base technology in question, the real challenge now becomes to focus on what features differentiate the product from its competitors. Same applies to software standards such as the ones of LIF and OGC.

The existence of a standard reduces design and production time, but that, in turn, should free the design team to work on improvements, that also in turn may become standard some day. The real business profit lies in this marginal period lasting from when new and distinguishing features appear until they become standards themselves. Of course, such strategic considerations will apply more or less to different products, but they are important all the same. For example, when developing the Product Finder application, it is not necessary to make the service "location-aware" (**Chapter 6**) as the "simple or basic" version in most cases will be as effective. However, a good strategic approach is to consider that "location-awareness" might become the norm or standard in all future LBS applications. As a result, adopting LIF's and OGC's standards early on might make sense.

Other factors can be addressed by the following question: Do you have the people with the experience and expertise necessary to utilize the standard? (If not, do you have the resources to develop them?). This is a basic question that applies to all the skills necessary to develop a product.

The next to last question to consider is extremely important and is the question of what will be the consequences of not adopting a given standard. Failure to adopt some standards can mean loss of profit as non-compliance will result in the application incapability to integrate with other applications. Therefore, IT managers need to work to make sure they are aware of the applicable standards in all cases, and to understand what will be the consequences of not adopting those standards.

The last issue we will discuss covers situations where either no standard exists, or current standards are out of date or unsatisfactory for one reason or another. In these situations the IT manager has several options: develop a proprietary solution to the problem without regard for how others are approaching the problem, join/create an industry consortium to tackle the problem, or submit a request to a standard body (i.e., ISO) to handle the problem. Creating a proprietary solution is an approach that we explore in strategy 1 of scenario 2 in **Chapter 6**, where a company (ImaHima) developed the Buddy Finder application. ImaHima was successful in locking-in customers that allowed the company to take advantage of a closed, proprietary standard to exploit the benefits of network externalities. Considering market expansion and opening up its standard, ImaHima might have considered submitting its standard to a standards body with the goal of making it an industry accepted standard.

8.4 Constraints, Implications, Consequences

The dimensions described earlier are general in nature. They could be used to describe the environment and options for a wide range of applications. In this section we describe the impacts of mobility and location-awareness on service adoption from a number of different perspectives. These cover social and behavioral impacts of LBS, as well as user privacy and security concerns.

The social and behavioral impacts of LBS deal with both the positive and negative consequences of LBS services and their implications for service adoption and diffusion. Current technology acceptance models, such as TAM, have many deficiencies for the range of contexts in which LBS are deployed. For example, current acceptance models can not explain the stark differences in the uptakes of WAP and i-mode service in Europe/U.S. and Japan respectively. In contrast it is important to integrate the perspectives of the user as a consumer, a network member and a technology user. This approach highlights the need to use specific technology adoption models for LBS at the individual level, but also raises the need to develop multi-perspective frameworks for understanding adoption behaviors at the group and organization levels[113].

In addition, more research is needed to explore how the factors influencing LBS adoption differs by level of analysis in the consumer market (e.g. families and non-work organizations) and how LBS adoption in the business segment impacts use in individuals' private lives and visa-versa. Services designers need to be able to understand the complex needs of all stakeholders in the overall social system of which the LBS is only a part. There may be mobility implications that a LBS is not particularly well suited to address with its current capabilities. For example while

---

[113] Jessop, L. and Robey, D., "The Relevance of Social Issues in Ubiquitous Computing Environments," Communicaitons of the ACM, vol. 45, pp. 88-91, 2002.

users frequently ask the OnStar service for directions to a particular store in a mall the service can currently only provide the mall's street address.

Privacy concerns may be particularly sensitive as services allow colleagues, family members or others to have real-time information on the location of individuals. Both privacy and security concerns could create resistance to LBS adoption[114]. At the same time a positioning capability is often used to increase security (e.g. tracking of children). In these applications location information provides a compelling value proposition. In both of these examples privacy and security is only maintained, however, if access to the location information is restricted to authorized users. How location information will be managed when the positioning capability becomes ubiquitous or seamless across all "world" (networks) is still uncertain[115].

As discussed in **Chapter 7**, some applications, location positioning is a minor value-add to the core service. In these cases the positioning capability may be so marginal that it could be excluded or made optional to counter privacy concerns. In short, applying these diverse perspectives on emerging indoor LBS applications provides insights into the types of users that are likely to adopt LBS, and major adoption barriers.

8.5 Conclusion

This thesis has demonstrated that in order for indoor LBS to become widely used, there is a need for both the infrastructure investment and the "killer" application (or at least a collection of sufficiently valuable applications). Without the LBS application the market will not invest in infrastructure, and without the infrastructure, the market for valuable LBS applications and their business models will not exist. The thesis distinguished four type of infrastructure: (1) communication, (2) positioning, (3) mapping, and (4) software (services).

Also, this thesis has argued that indoor LBS applications will need more modularity and standardization across these infrastructures in order to reach critical mass. Also, there is a definite need for LBS for the "indoor world," but it will take new users a while to become comfortable with such technology. The area shows promise, but progress has been limited by the "closed" and "self-contained" aspects. The individual killer application need to be bundled (via Web services) to counter the lack of services with a high added value to the end users and to reach critical mass.

The dimensions, filters, and constraints discussed in this concluding chapter should give the IT manager and service designer the background necessary to make the most appropriate decision regarding the overall service deployment strategy. Each choice has its own set of costs, risks, rewards, and benefits. The manager needs to consider the resources available to the company, weigh them against the costs involved with each process. Following these guidelines should also

---

[114] Warrior J., McHenry, E., and McGee, K. "They Know Where You Are," Spectrum, IEEE, vol. 40, pp. 20-25, 2003.

[115] Snekkenes, E. "Concepts for Personal Location Privacy Policies," presented at EC'01, Tampa, 2001.

allow the IT manager to make the most appropriate choice when it comes to evaluating standards for adoption in development projects.

The last filter focused on key standards issues, such as who should support standards, what standards efforts are ripe for whom now, and what will likely have to happen first to promote/trigger more interest in standards. **Figure 8.5** summarizes what standards apply to which infrastructure type. As demonstrated through out this thesis, the issue of interoperability needs should be addressed across all of the infrastructure types.

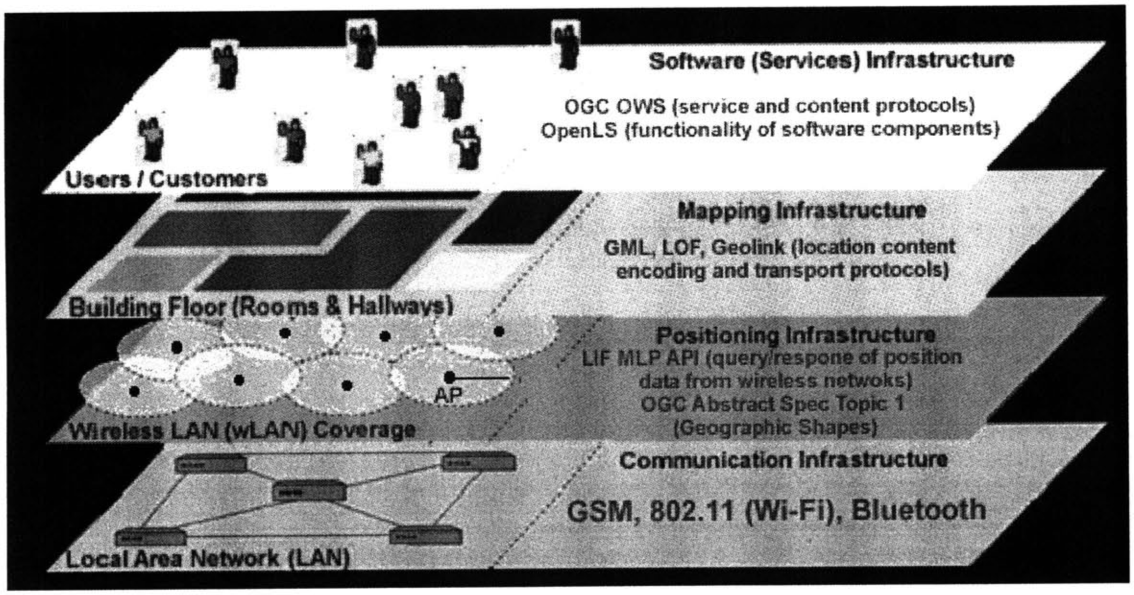


Figure 8.5: Infrastructure Types and Applicable Standards

And, **Figure 8.6** portrays the LBS value-chain spectrum and where OpenGIS/OpenLS and LIF standards apply with respect to the different types of infrastructure providers. Mainly, OpenGIS/OpenLS standards apply to the mapping and software (services) infrastructure types. LIF applies to the communication and positioning infrastructure.

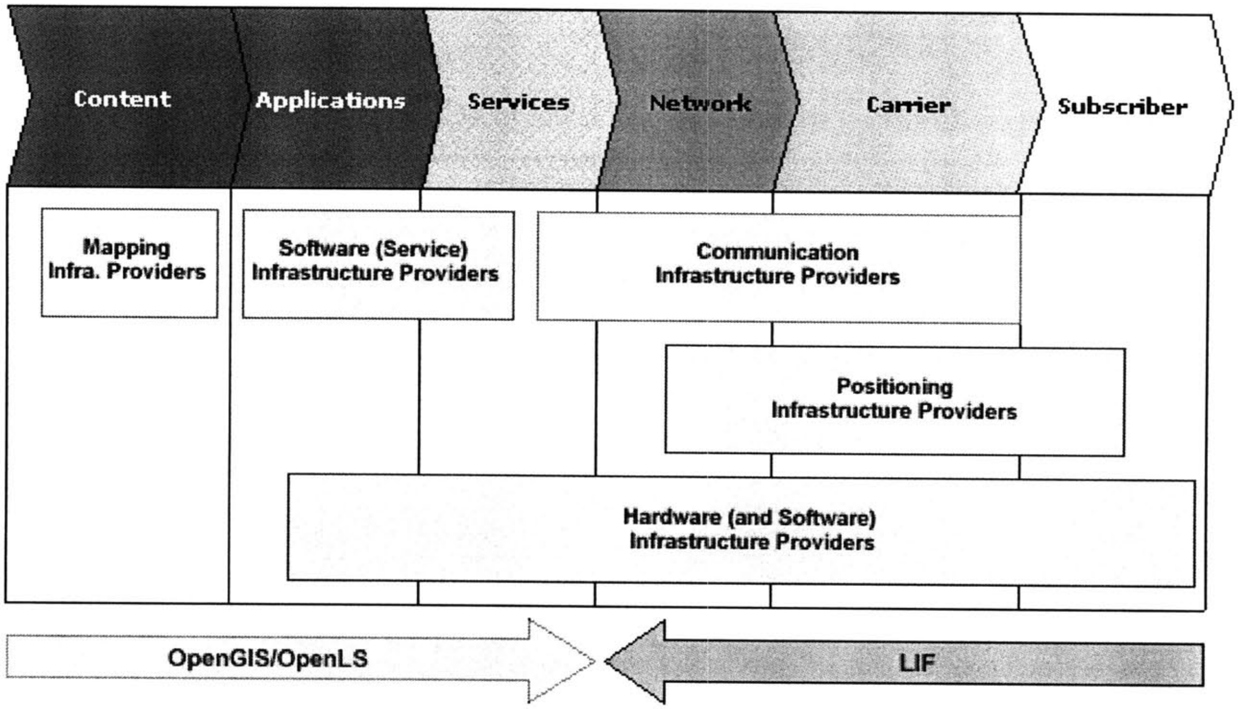


Figure 8.6 LBS Value-Chain, Infrastructure Types, and Applicability of Standards

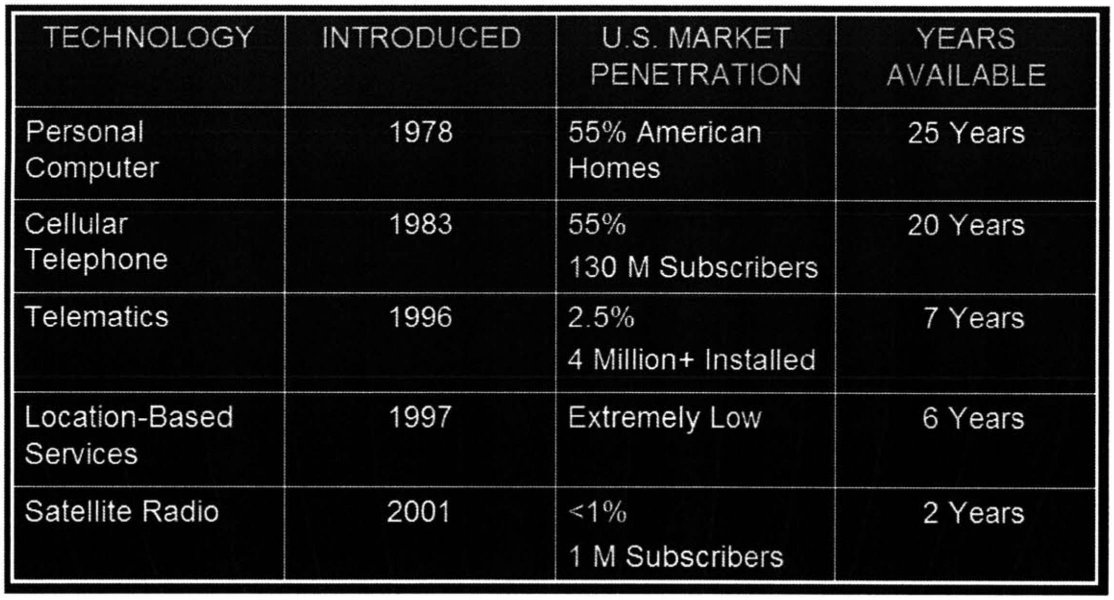
8.6 The Potential and Vision

Imagine walking down a hall in a shopping mall, an airport, etc. Instead of having boring advertisements that don't pertain to one's interests, the futuristic approach as portrayed in Steve Spielberg's movie, "Minority Report," would be to have advertisements specifically targeted to one's personal profile.

The positioning technology that senses objects approaching a certain area (e.g.., a particular hallway in a shopping mall) to allow this type of a personalized service already exists today, such as the ActiveBadges LPS (**Chapter 4**). Routing schemes can be calibrated in order to obtain the desired delay. The user's movements can be tracked in order to put relevant information as near as possible to his/her location to reduce the wireless link congestion. It is also possible to model the user's future behavior in order to reduce the expected network load by distributing information along his/her possible path and by pre-fetching data (which will be likely requested by the user in a future time) under good radio link conditions if substantial degradation is foreseen along the modeled user path, resulting in faster perceived service. Of course, a crowded mall would easily overload such a system if the result were shown on video walls or shared hollograms.

Figures 8.7: Steven Spielberg's "Minority Report" – Futuristic Shopping Mall

However, we might be decades away from such a vision. **Figure 8.8** shows how many years it took for market penetration for technologies such as a cellular phone and LBS (outdoor) services.



TECHNOLOGY	INTRODUCED	U.S. MARKET PENETRATION	YEARS AVAILABLE
Personal Computer	1978	55% American Homes	25 Years
Cellular Telephone	1983	55% 130 M Subscribers	20 Years
Telematics	1996	2.5% 4 Million+ Installed	7 Years
Location-Based Services	1997	Extremely Low	6 Years
Satellite Radio	2001	<1% 1 M Subscribers	2 Years

Figure 8.8: Timelines Required to Achieve Market Penetration (source: Driscoll-Wolfe)

In order for the Minority Report vision to become practical (for better or worse), a rich set of interrelated standards and services will need to mature. Engagement of industry standards organizations with universities, government and the public will be needed to insure early attention to the standard component middleware for authentication, data control, and customization that will be needed for seamless, layered location based services to exploit the envisioned potential while being trusted by and affordable to the public.

REFERENCES

Addlesee, M.D., Jones, A., Livesey, F., Samaria, F.: The ORL Active Floor. IEEE Personal Communications, October 1997, Vol. 4 No. 5.

Adjie-Winot, W., Schwartz, E., Balakrishnan, H., and Lilley, J. The design and implementation of an intentional naming system. In Proc. ACM Symposium on Operating Systems Principles (Kiawah Island, SC, Dec. 1999), pp. 186–201.

Applegate, Lynda M., F. Warren McFarlan, and James L. McKenney (1999). Corporate Information Systems Management: Text and Cases, 5th ed., Irwin/McGraw-Hill.

Asthana, A., Cravatts, M., and Krzyzanowski, P. An indoor wireless system for personalized shopping assistance. In Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pages 69-74, Santa Cruz, California, December 1994. IEEE Computer Society Press.

Bauer, M., Becker, C., and Rothermel, K. Location models from the perspective of context-aware
applications and mobile ad hoc networks. In Workshop on Location Modeling for Ubiquitous Computing (2001).

Beigl, M., Zimmer, T., and Decker, C. "A location model for communicating and processing of context," Personal and Ubiquitious Computing, pp. 341-357, 2002.

Boswell, R. Location-Based Technology Pushes the Edge. Telecommunications, June 2000.

Brumitt, B., and Shafer, S.. Topological world modeling using semantic spaces. In Workshop Proceedings of Ubicomp 2001.

Brumitt, B., Meyers, B., Krumm, J., Kern, A., and Shafer, S (2000). EasyLiving: Technologies for intelligent environments. In Proceedings of Second International Symposium on Handheld and Ubiquitous Computing, HUC 2000, pages 12-29, Bristol, UK, September. Springer Verlag.

Burnett J.(2000) OGC Announces Wireless Location Services Initiative, http://www.openls.org/News/00-10-30.htm

Coulouris, G., Mitchell, S., Naguib, H. "Middleware Support for Context-Aware Multimedia Applications", Proceedings of the third International Working Conference on Distributed Applications and Interoperable Systems, IFIP Conference Proceedings, vol 198, pp 9-22, September 2001

Czerwinski, S., Zhao, B., Hodes, T., Joseph, A., and Katz, R. An Architecture for a Secure

Service Discovery Service. In Proc. 5th ACM MOBICOM Conf. (Seattle, WA, Aug. 1999), pp. 24–35.

Dey, A., Futakawa, M., Salber, D., and Abowd, G. The Conference Assistant: Combining Context-Awareness with Wearable Computing. In Proceedings of the 3rd International Symposium on Wearable Computers (ISWC '99), pages 21-28, San Francisco, CA, October 1999. IEEE Computer Society Press.

Fabrizi, E. and Sa.otti, A. (2002). Augmenting topology-based maps with geometric information, Robotics and Autonomous Systems 40(2-3): 91-97.

Flury,T., and Privat, G. An infrastructure template for scalable location-based services. France Télécom R&D, DIH/OCF 2003.

Hendrey, G. Managing the wireless internet. RF Design, March 2001, p50-56.

Hecht L. (2001) Open Location Services: Vision and Objectives, Corporate Roles and Work Program Summary, http://www.openls.org/docs/OLS_Oveview.htm

Hightower, J. and Borriello, B, "Location Systems for ubiquitous computing", IEEE Computer, Aug 2001.

Horswill,I. (1998). The Polly system, in D. Kortenkamp, R. P. Bonasso and R. Murphy (eds), Arti.cial Intelligence and Mobile Robots: Case studies of successful robot systems, MIT Press, Cambridge, M A, chapter 5, pp. 125-139.

Horvitz, E. Principles of Mixed-Initiative User Interfaces. In Proceedings of CHI'99, ACM SIGCHI Conference on Human Factors in Computing Systems, May 1999.

Krechmer, K (1999). The Principles of Open Standards. Standards Engineering Society World Standards Day Paper Competition, second place. (http://www.ses-standards.org/library/krechmer.pdf)

Krogstie, J., Lyytinen, K., Opdahl, A., Pernici, B., Siau, K., and Smlolander, K.., "Mobile Information Systems - Research challenges on the conceptual and logical levels," presented at ER/ISIP 81. Workshop MobIMod'2002, Finland, 2002.

Lamming, M., Flynn, M. "Forget-me-not"—Intimate Computing in Support of Human Memory. Proceedings of FRIEND21, International Symposium on Next Generation Human Interface, Meguro Gajoen, Japan, 1994.

Leonhardt, U., Magee, J., Dias, P. "Location service in mobile computing environments", Computers and Graphics, vol 20 n° 5, pp 627-632, 1996

Leonhardt, U., Magee J. (1998) "Multi-Sensor Location Tracking", Proceedings 4th ACM/IEEE Int'l Conf. Mobile Comp, pp 203-214, October, 1998

Liao, L., Fox, D., Hightower, J., Kautz, H., and Schulz, D. Voronoi Tracking: Location Estimation using Sparse and Noisy Sensor Data. Proceedings of the International Conference on Intelligent Robots and Systems (IROS), IEEE/RSJ, 2003.

Kuipers,B. J. and Levitt, T. S. (1988). Navigation and mapping in large-scale space, AI Magazine
9(2): 25-13

Kuipers,B. J., Froom,R ., Lee,W .-Y. and Pierce, D. (1993). The semantic hierarchy in robot learning, in J. Connell and S. Mahadevan (eds), Robot Learning, Kluwer Academic Publishers, Boston, MA, pp. 141-170.

Koenig, S . and Simmons, R. G. (1998). Xavier: A robot navigation architecture based on partially observable markov decision process models, in D. Kortenkamp, R. P. Bonasso and R. Murphy (eds), Artificial Intelligence and Mobile Robots: Case studies of successful robot systems, M IT Press, Cambridge, MA, chapter 4, pp. 91-122.

McCarthy, J. F. and Anagnost, T. D. EventManager: Support for the Peripheral Awareness of Events. In P. Thomas and H. W. Gellersen, editors, Handheld and Ubiquitous Computing, number 1927 in Lecture Notes in Computer Sciences, page 227 to 236. Springer Verlag, Germany, September 2000.

Nicklas, D., Mitschang, B. The NEXUS Augmented World Model: An extensible approach for mobile, spatially aware applications, 7th International Conference on Object-Oriented Information Systems, 2001

Nourbakhsh, I., Powers, R. and Birchfield, S . (1995). DERVISH: An office navigating robot, AI Magazine 16(2): 53-10.

Nourbakhsh, I . (1998). Dervish: An once-navigating robot, in D. Kortenkamp, R. P. Bonasso and R. Murphy (eds), Artificial Intelligence and Mobile Robots: Case studies of successful robot systems, MIT Press, Cambridge, M A, chapter 3, p p. 73-10.

OpenGIS Catalog Service Specification: http://www.opengis.org/techno/specs/99-051.pdf

OpenGIS GML Specification: http://www.opengis.org/techno/specs/00-029/GML.html

OpenGIS Style Layer Descriptor (SLD) Specification: http://www.opengis.org/techno/discussions.htm

OpenGIS WMS Specification: http://www.opengis.org/docs/01-068r2.pdf

OpenGIS WFS Specification: http://feature.opengis.org/members/archive/arch01/01-023r1.pdf

Poncela, A., Perez, E. J.,Ban dera, A.,U rdiales,C . and Sandoval, F. (2002). Efficient integration of metric and topological maps for directed exploration of unknown environments, Robotics and Autonomous Systems 41(1): 21-29.

Samet, H. (1984). "The Quadtree and Related Hierarchical Data Structures," Association for Computing Machinery Computing Surveys, 16

Samet, H.: The Design and Analysis of Spatial Data Structures, Addison-Wesley, Reading, MA, 1990, ISBN 0-201-50255-0.

Schiele, B., Pentland, A.: Probabilistic Object Recognition and Localization. In ICCV'99 International Conference on Computer Vision, Greece, September 1999.

Schilit, B., Adams, N., Want, R.: "Context-Aware Computing Applications", IEEE Workshop on Mobile Computing Systems and Applications.

Schroter, D., Beetz, M., and Gutmann, J-S. RG Mapping: Learning Compact and Structured 2D Line Maps of Indoor Environments. IEEE Intternational Workshop on Robot and Human Interactive Communication, September, 2002.

Spiteri, M.D., Bates, J., "An architecture to support storage and retrieval of events", Proceedings of Middleware 98, IFIP International Conference on Open Distributed Processing, Lancaster, UK, September 1998.

Steen, M. v., Hauck, F., Homburg, P., and Tanenbaum, A.: Locating Objects in Wide-Area Systems, IEEE Communications Magazine, pp. 2–7, 1998.

Thrun, S. and Bücken, A. Integrating Grid-Based and Topological Maps for Mobile Robot Navigation. Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI), Portland, Oregon. 1996.

"T-Mobile Taps Boingo To Integrate Wi-Fi, GPRS": http://www.wi-fiplanet.com/news/article.php/2115671

Tomatis, N .,Nourbakhsh, I . and Siegwart, R. (2003). Hybrid simultaneous localization and map building: a natural integration of topological and metric, Robotics and Autonomous Systems 44 (1): 3-4.

Veizades, J., Guttman, E., Perkins, C., and Kaplan, S. Service Location Protocol, June 1997.

Willingmyre, George T. International Standards at the Crossroads. ACM StandardView 5(4), December 1997, p190-194 (http://www.acm.org/pubs/articles/journals/standardview/1997-5-4/p190-willingmyre/p190-willingmyre.pdf)

Websites

Auto ID Center http://www.autoidcenter.org

Driscoll-Wolfe Wireless Internet Location Services Study: http://www.driscoll-wolfe.com/

Driscoll C. (2001) "Are Consumers Interested in Wireless Internet Location-Based Services?"http://special.northernlight.com/wireless/w_locationbased.htm

Estrin diffusion: http://www.cens.ucla.edu/Estrin/index.shtml

IETF: http://www.ietf.org/

Jini (TM). http://java.sun.com/products/jini/

LIF /OMA: http://www.openmobilealliance.org

MIT Cricket http://nms.lcs.mit.edu/projects/cricket/

MIT's INS by Balakrishnan: http://wind.lcs.mit.edu/

MS Easy Living project: http://research.microsoft.com/easyliving/

MSN Shopping Assistant: http://shopping.msn.com/softcontent/softcontent.aspx?scpId=3565&scmId=1422

OpenGIS Consortium, Inc. (OGC): http://www.opengis.org

OGC's OpenLS Initiative: http://www.openls.org

OpenGIS Web Services (OWS): www.opengis.org/ows

ISO TC/211 www.isotc211.org

US Federal Communications Commission's E911 telecommunication initiatives: http://www.fcc.gov/e911

3G: http://www.3g.co.uk/PR/Oct2003/5950.htm

3GPP: www.3gpp.org

Segway (the human transporter) http://www.segway.com/

Sentient Computing Project: http://www.uk.research.att.com/spirit/

UWB: http://www.uwb.org

Ubiquitous Systems: http://www.ubiquitous-systems.com

WAP Forum: www.wapforum.org

W3C: http://www.w3c.org/

ZigBee Alliance website http://www.zigbee.org/

APPENDIX A

INDOOR LOCATION POSITIONING SYSTEMS

Featured Indoor Local Location Positioning Systems

System 1: ActiveBadge (AT&T Cambridge)[116]

The Active Badge system was one of the earliest indoor systems for position information. Badges can be worn by people or attached to equipment (i.e., computers). Location is derived by adding cell information of the IR network to the data. It tracks objects having an IR badge attached to it, which periodically transmits (every 10 seconds) its unique ID using IR transmitters. Fixed (known positions) IR receivers pick up this information and relay it over a wired network to the location manager software.

The system typically allows for room-level resolution since IR waves cannot pass through walls. A particular badge is associated with the fixed location of the receiver that hears it. When combined with low-energy radio fields, the system can also provide more accurate measurements. IR networks have the great advantage of not requiring a part of the radio spectrum. On the other hand, the short effective range of the transmitters makes comprehensive deployment impractical. The range of the minimalist IR beckon implementation ranges from 3 to 10 meters, allowing a footprint from 1 m² to about 20 m². The IR beckon signals are IrDA standard compliant

Possible applications include telephone call re-routing, GUI teleporting, visualize people's location on a computer, location-sensitive communications, location-oriented paging, security and environmental control[117].

A badge's location is relative, representing, for example, the room—or other IR constraining volume—in which the badge is located. As far as the data model is concerned, the unit of location is a tuple consisting of badge address, location address (i.e., room number), and timestamp. The location of an object is modeled as a dynamic attribute of the object, which is implemented by a pointer to a service interface from that object. Since the system is tied to its sensors, abstraction is lacking.

The system also demonstrates one of the first large distributed software architectures for handling location data, which in this case is symbolic[118]. Its client-server approach architecture

---

[116] Active Badge website: http://www.uk.research.att.com/ab.html

[117] Elrod, S., Hall, G., Costanza, R., Dixon, M., Rivieres, J. *Responsive Office Environments*. Communications of the ACM, July 1993.

inspired future generations including the Bat system and PinPoint's system. In these architectures, the hardware tag attached to mobile devices is active, and responds to queries from a central controller and location database about its whereabouts.

The system functionality is portioned into services provided by the following servers:

- The Location server collects the badge sightings measurements from fixed IR sensors around the building, aggregates the data into a central location database, and provides an API for applications to take advantage of the data. It maintains a cache of the last sightings, which consists of the badge address, the sensor address, and a time-stamp.
- The Name server offers a White Pages directory service that maps badge addresses and location addresses into more detailed information, such as the name of the bearer.
- The Message server co-ordinates into more detailed information, such as the name of the bearer.
- The Exchange server controls the federation of location service between organizations. It encapsulates the issues of security, access control, and information exchange between administrative domains.

While the system provides accurate room scale location information it suffers from several drawbacks:

1. It scales poorly due to the limited range of IR (the system is targeted towards a federation of small to medium sized spaces, hence, the issue of scalability has not been addressed sufficiently).
2. It incurs significant installation and maintenance costs
3. IR suffers from dead-spots caused by fluorescent lighting or direct sunlight, which is likely to be a problem in rooms with windows. Diffuse IR has an effective range of several meters, which limits cell sizes to small- or medium-sized rooms. In larger rooms, the system can use multiple IR beckons.
4. The object tracking nature of the system may introduce privacy concerns among users.

System 2: Active Bat (AT&T Cambridge)[119]

The Active Bat system[120], which uses a combination of RF and ultrasound time-of-flight to estimate the distance, augmenting the previous generation Active Badges to provide more accurate positioning in certain circumstances – it can locate objects to within 9 cm of their true position (95% probability). An application example is the Visual Tracking Service, which is proactive. Having a room equipped with several cameras, the Bat system is then used to track a user using the nearest camera to keep the user in-shot. Thus, the system can create a video stream tracking the user as he/she walks around in the room. Other proactive applications are routing incoming telephone calls to the nearest telephone and having computer desktops displayed at the workstation closest to the user.

---

[118] Want, R., Hopper, A., Falcao, V., and Gibbons, J. The Active Badge Location System. *ACM Transactions on Information Systems 10*, 1 (January 1992), 91–102.

[119] Active Bat website: http://www.uk.research.att.com/bat/

[120] A. Ward, A. Jones, and A. Hopper, "A new location technique for the active office," IEEE Personal Communications, vol. 4, pp. 42–47, Oct. 1997.

The system consists of a collection of mobile or fixed wireless transmitters, a matrix of receiver elements, and a central RF base station. Objects are tagged with wireless transmitter ("bats"), each of which has a FPGA, an ultrasound transmitter, a microprocessor, a radio transceiver and has a unique ID associated with it. The bats emit periodic ultrasonic signals to receivers mounted throughout the ceiling, which are connected together by a serial wire network to form a matrix. The bats are located by a location database (central controller) and the world model stores the correspondence between bats and their owners, applying type-specific filtering algorithms to the bats location data to determine the location of the object which owns it[121]. From the times of flight of the pulse from the bat to each of the receivers that detected it, the system can calculate the 3D position of the bat, to an accuracy of about 3cm.

The system derives its high accuracy from a tightly controlled and centralized architecture that tracks users and objects. However, at the expense of this, user privacy is of concern. Using ultrasound requires a large fixed-sensor infrastructure throughout the ceiling and is rather sensitive to the precise placement of these sensors. Thus, scalability, ease of deployment, and cost are disadvantages of this approach.

The system can also compute orientation information given predefined knowledge about the placement of the bats on the rigid form of an object and allowing for the ease with which ultrasound is obstructed. Each interface for a serial data network has a GUID for addressing and recognition.

An extension to the Active Badge system allows equipment to be tracked using a low-power version of the Badge called an *Equipment Tag*[122]. The developers describe a 'nearest printer' service offered to users of portable computers. Tags placed on the computer and printers report their positions, and the computer is automatically configured to use the nearest available printer as it is moved around a building.

System 3: RADAR (Microsoft)[123]

RADAR[124] is an in-building location-aware tracking system based on the 802.11 WaveLAN wireless networking technology. It allows RF wireless-LAN-enabled mobile devices to compute their location based on the signal strength of known infrastructure access points (AP). The system leverages an existing RF infrastructure that provides the building's general-purpose wireless networking, without setting up an additional location tracking components.

The system calculates the 2D position coordinates of a device either by empirical methods based on comparison (fingerprinting) with previous measured locations mapped on the Radio Map or using a mathematical model of indoor radio signal propagation (the map is measured in advance

---

[121] Harter, A., and Hopper, A. A New Location Technique for the Active Office. *IEEE Personal Communications 4*, 5 (October 1997), 43–47

[122] Harter, A., Hopper, A. *A Distributed Location System for the Active Office*. IEEE Network, Special Issue on Distributed Systems for Telecommunications, January 1994.

[123] RADAR website: http://research.microsoft.com/~bahl/MS_Projects/projects.html#radar

[124] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of IEEE INFOCOM 2000*,pp. 775–784, Mar. 2000.

for its radio propagation properties). Hence, the system implements a location service utilizing the information obtained from an already existing RF data network[125]. The system is able to estimate a user's location to within 2-3 meters (about the size of a typical office room) of his/her actual location (with 50% probability). This is achieved using RADAR's scene-analysis. Another implementation of RADAR uses lateration, which has a 4.3-meter accuracy at the same probability level. Although the scene-analysis version provides greater accuracy, significant changes in the environment, such as moving metal file cabinets or large groups of people congregating in rooms or hallways, may necessitate reconstructing the predefined signal-strength database or creating an entirely new database.

In addition to user's location, RADAR can also record the direction/orientation (one of north, south, east, or west) that the user is facing at the time the measurement is made. Also, RADAR allows for the user to indicate his/her current location by clicking on a map of the floor. The user's coordinates (x,y) and timestamp are recorded.

In case privacy is a concern, the architecture of RADAR enables a mobile device to track its own location silently without other nodes in the system being aware of it. In the extreme, a mobile device can essentially turn off data connectivity and use its wireless interface (in conjunction with RADAR) solely for the purpose of tracking its own location. Other than the signal strength values derived from beckons, the mobile device only needs the Radio Map and the layout map of the building, which it can download say the first time it enters the building.

Some of RADAR's limitations are the following:

1. Object tracking – it must support a wireless LAN (the NIC card), which may be impractical on small or power-constrained devices.
2. Effect of multi-floored buildings (or three dimensions). Signal aliasing between points on adjacent floors could cause the system to place the user on the wrong floor.
3. Signal generated by the mobile host might be obstructed by the user's body. While this may not be realistic given the antenna design and positioning for existing wireless LANs, it may be possible to approximate an "ideal case" with new antenna designs (e.g., omni-directional wearable antenna).

System 4: Cricket (MIT)[126]

Cricket uses a combination of RF and ultrasound signals in a decentralized, uncoordinated architecture[127]. Wall- and ceiling-mounted beckons, are spread through the building, publishing location information on an RF signal. With each RF advertisement, the beckon transmits a concurrent ultrasonic pulse. The mobile device (the listener) receives these RF and ultrasonic signals, correlates them to each other, estimates distances to the different beckons using the difference in RF and ultrasonic signal propagation times, and therefore, infers the distance and

---

[125] Bah, P., and Padmanabhan, V. RADAR: An In-Building RF-based User Location and Tracking System. In *Proc. IEEE INFOCOM* (Tel-Aviv, Israel, March 2000).

[126] MIT Cricket website: http://nms.lcs.mit.edu/projects/cricket/

[127] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *MOBICOM 2000*, pp. 32–43, Aug. 2000.

space they are currently in. Cricket achieves 1-3m accuracy (with portion-of-a-room granularity of 4x4 feet).

In addition, by separating the processes of tracking services and obtaining location information, multiple resource discovery systems can be accommodated. Applications learn about services in their vicinity via an active map that is sent from a map server application, and interact with services by constructing queries for services at a required location[128]. Once location information is obtained, services advertise themselves to a resource discovery service such as the MIT Intentional Naming System (INS)[129], IETF Service Location Protocol[130], Berkeley Service Discovery Service[131], or Sun's Jini discovery service[132]. These resource discovery services handle service and device mobility within the naming system. Cricket applications include location-aware applications that enable users to discover resources in their physical proximity[133], active maps that automatically change as a user moves[134], and applications whose user interfaces adapt to the user's location.

Most of the other systems are based on a cellular approach, in which either the mobile device detects its cell or the system determines which mobile devices are in each cell. Being a location-support system (rather than a conventional location-tracking system), Cricket does not track and store location information for services and users in a centrally maintained database. It helps devices learn where they are and lets them (applications) decide whom to advertise this information to. As a result, user-privacy concerns are adequately met.

In contrast, systems like the Bat have the central controller know where each wall- or ceiling mounted device is located. This has two disadvantages. First, user-privacy is compromised because a listener now needs to make active contact to learn where it is (in Cricket, a listener is completely passive). Second, it requires a centrally managed service, which does not suit Cricket's autonomously managed environment particularly well. Cricket's beckons advertising location information are self-contained and do not need any infrastructure for communication amongst themselves.

Some of the limitations of the system are the following:

1. Lack of centralized management or monitoring.
2. Computational burden (and consequently power burden) that timing and processing both the ultrasound pulses and RF data place on the mobile receivers.

---

[128] Adjie-Winoto, W., Schwarz, E., and Balakrishnan, H., and Lilley, J. The design and implementation of an intentional naming system. In *Proc. ACM Symposium on Operating Systems Principles* (Kiawah Island, SC, Dec. 1999), pp. 186–201.

[129] MIT INS: http://nms.lcs.mit.edu/projects/ins/index.html

[130] Veizades, J., Guttman, E., Perkins, C., and Kaplan, S. *Service Location Protocol*, June 1997. RFC 2165 (http://www.ietf.org/rfc/rfc2165.txt).

[131] Czerwinski, S., Zhao, B., Hodes, T., Joseph, A., and Katz, R. An Architecture for a Secure Service Discovery Service. In *Proc. 5th ACM MOBICOM Conf.* (Seattle, WA, Aug. 1999), pp. 24–35.

[132] Jini. http://java.sun.com/products/jini/, 1998.

[133] Harter, A., Hopper, A., Steggles, P., Ward, A., and Webster, P. The Anatomy of a Context-Aware Application. In *Proc. 5th ACM MOBICOM Conf.* (Seattle, WA, Aug. 1999).

[134] Schilit, B. and Theimer, M. Disseminating Active Map Information to Mobile Hosts. IEEE Network (Sep/Oct 1994), 22-32

System 5: MobileShadow (SafeSoftware)[135]

The MobileShadow system is a distributed software infrastructure for proactive location-aware services and applications. It is built upon an 802.11b wLAN and uses the cell-based approach for its location system. Each user has a virtual alter ego called user agent. This user agent always resides at the same location as the user. Each user agent can interact with other user agents and local information such as reminders, advertisements, and local messages. The system is optimized to answer queries such as who is nearby the building exit.

In addition to proactive location-awareness, the system also aims scalability and adaptivity. The decentralized infrastructure and the use of mobile code technology tackles the scalability issues. And the use of mobile agents and a component-based design support adaptivity by manipulation of the user. Moreover, adaptivity is used as adaptation by manipulation – the user is enabled to easily manipulate a significant part of the system.

Related research projects mainly concentrate on the locating issues such as Cricket, Cyberguide, Active Bat, EventManager (Accenture), or active badge. Each system usually has one proof-of-concept application. The most closely related system is the stick-e document approach[136]. Another closely related project is the Lancaster Tour Guide[137].

System 6: UbiTags (UbiSense)[138]

UbiSense uses a network of sensors installed into a building's existing communication network. The sensors use UWB to detect and react to the position of Ubitags. Ubitags are carried by people or attached to assets and are in constant contact with the network of sensors. The sensors send the Ubitag location information to the Ubisense software solution, which creates a detailed, real-time view of the environment. This model can be used by an unlimited number of simultaneous programs that are able to respond immediately to changes in the building environment.

Location positioning systems based on conventional RF technology work poorly indoors because they are plagued by multipath distortion caused by RF signals reflected from walls, desks, people and equipment. This can often lead to positioning errors of several meters. UWB radio systems can be accurate to about 15cm indoors because they are much less affected by multipath distortion than conventional RF systems and the calculation is based on time of arrival (TOA) rather than signal strength. An advantage of UWB over systems that use IR technology is in its ability to pass through objects such as walls and clothing. One common problem with IR systems is that they will fail even if the tag is covered up by a shirt or jacket.

---

[135] MobileShadow website: http://www.mobileshadow.net/welcome.html

[136] P. Brown. The stick-e document: a framework for creating context-aware applications. Electronic Publishing, 8(2&3):259 to 272, Jun & Sep 1995.

[137] N. Davies, K. Cheverst, K. Mitchell, and A. Efrat. Using and Determining Location in a Context-Sensitive Tour Guide. IEEE Computer, 34(8):35 to 41, August 2001.

G. D. Abowd, C. G. Atkeson, J. Hong, S. Long, R. Kooper, and M. Pinkerton. Cyberguide: A Mobile Context-Aware Tour Guide. Baltzer/ACM Wireless Networks, 3(5):421 to 433, October 1997.

[138] UbiSense website: http://www.ubisense.net/

The proprietary software platform is really what sets the Ubisense system apart from other companies developing UWB-based tracking systems.

System 7: Real Time Location System (RTLS) (PinPoint Corp)[139]

The Real Time Location System (RTLS)[140] uses RF signals that activate transponders attached to the object. The system is composed of cells within a building and uses RF signals and multiple antennas (up to 16) at the cell controller to process the signal from a tag. Time of flight (TOF) is measured to calculate an object's position using PinPoint Cell Controllers. The system's locating ability varies depending on the number of antennas installed in an area. In general, it can locate and track objects based on the systems' readers that emit a RF code and phase from a 30-meter distance with a 1-3 meter accuracy (granularity of about 10 meters).

RTLS will let you:

- Instantly locate and asset or person.
- Maintain a complete log of movements for auditing, security, and usage analysis.
- Generate an instant inventory of all tagged assets.
- Trigger alerts if a tag leaves or enters specific areas.
- Monitor and control access to and movement of assets and personnel.
- Integrate tracking/ finding applications in a single, inexpensive network infrastructure

RTLS is similar to both SpotON and the RADAR. It has decent accuracy and is somewhat scalable but has the disadvantage of being very expensive. Also, while the Bat system uses a combination of RF and ultrasound to estimate distance, RTLS uses spread-spectrum RF and multiple antenna at the controller to process messages from a tag. By virtue of being a commercial product, RTLS offers easier deployment and administration than many research systems.

Some of the systems limitations are the following:

1. It requires specialized hardware to do location determination.
2. Each antenna has a narrow cone of influence, so that ubiquitous deployment becomes prohibitively expensive. Thus, RTLS best suits large indoor space settings such as hospitals or warehouses.
3. Difficulties arise when interoperating with the 802.11 wireless networking infrastructure because of radio spectrum collision[141] (unregulated in the Industrial, Scientific, and Medical band).

---

[139] PinPoint website: http://www.pinpointco.com

[140] Werb, J., and Lanzl, C., "Designing a positioning system for finding things and people indoors," *IEEE Spectrum*, vol. 35, pp. 71–78, Sept. 1998.

[141] Hightower, F., and Borriello, G. Location system for ubiquitous computing. *IEEE Computer*, 34(8):57–66, August 2001.

Other LPS Systems

System 8: Constellation (Isense)[142]

The Constellation tracking system uses a combination of accelerometer gyros and ultrasonic sensors to estimate position and orientation. It relies on an active set of ultrasonic beckons to determine the initial tracking position of the device and then recursively refines the orientation estimation using information gathered by the inertial sensors. Specifically, the IS-900 motion tracker, tracks the position and orientation of objects with high precision (0.05 degrees and ~1 mm) at high rates (180 Hz, ~2 ms latency). This type of data is useful for virtual reality applications, including interactive visualization in projection environments and simulation/training systems with head-mounted displays.

The tight coordination that is required between the receivers and transmitter of this system makes it unsuitable for large-scale indoor deployment. Even if the tracking area is scalable, unlike the other kinds of positioning systems reviewed, Constellation requires the installation of infrastructure in each room that needs tracking. As a result, in practice, most users (customers) are tracking less than a couple thousand square feet of area. Also, it is also unclear that this can be implanted in a handheld-like form factor.

Constellation is similar in its basic principles of operation to an aided inertial navigation system, except that it operates indoors, has much finer resolution and accuracy, and uses acoustic rather than RF technology for range measurements.

The system is configured for tracking an HMD (Head Mounted Display) in a wide-range VR or AR application. The HMD is equipped with an integrated inertial sensing instrument called the InertiaCube and ultrasonic range-finder modules (URMs). The range-finder modules communicate with a constellation of transponder beckons, which may be mounted at any known locations in the environment.

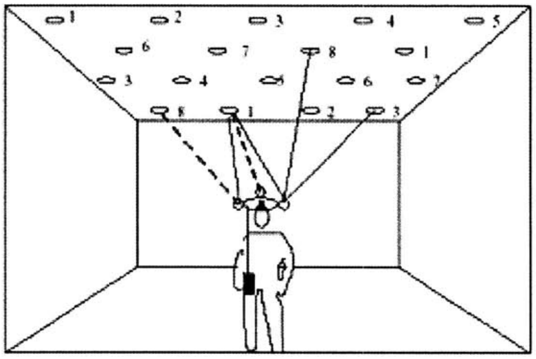


Figure A.1: Constellation Head Mounted Display

---

[142] Constellation website: http://www.isense.com

The API consists of a .dll (or .so) that obtains data from the tracker over a serial port or Ethernet connection. To use the .dll in a customer application to get data is simply a matter of a function call to initialize the system, and then calling a getData function each time the app requires the current pose data. The API has many more complex functions for setting up the constellation and configuring the tracker performance, but most people use the utility program to do all the configuration and save it in the tracker. As a result, their application has a very simple interface.

This system could not be used for the kinds of applications that use Wi-Fi, for example, because it is more expensive and those application don't require the precise tracking capability. This product is really intended for wearable augmented reality applications.

System 9: HiBall

The HiBall system uses opto-electronic tracking of hundreds or thousands of IR LEDs mounted in special ceiling panels[143]. It provides rapid updates of receiver position and orientation, but requires the installation of large arrays of LEDs in the ceiling and carefully machined camera at the client, which will significantly increase deployment costs.

Systems 10: MotionStar (Ascension) [144]

MotionStar (similar to Startrak[145] and Aurora[146]) is a commercial magnetic motion tracker used in virtual reality and simulation applications such as head-mounted displays and biomechanic motion capture. Electro-magnetic sensing offers a classic position tracking method by sending magnetic pulses and detecting the change of field strength along three orthogonal axes [147]. They provide very high precision and accuracy, on the order of less than 1 mm spatial resolution, 1 ms time resolution, and 0.1 orientation capability. These systems usually require a centralized coordination between the magnetic transmitters and receivers and are susceptible to magnetic interference from presence of metals or other conductive materials in the environment, which causes problems in many indoor environments.

Ascension offers a variety of motion-capture solutions, including Flock of Birds the MotionStar DC magnetic tracker[148]. These tracking systems are based on pulsed DC magnetic fields from a transmitting antenna in a fixed location. Multiple sensors are placed on body mounted peripherals, such as data gloves, and their output is processed to determine a person's location and orientation with a high degree of precision. The system computes the position and orientation of the receiving antennas by measuring the response in three orthogonal axes to the transmitted field pulse, combined with the constant effect of the earth's magnetic field.

---

[143] Welch, G. Bishop, G., Vicci, L. Brumback, S., Keller, K., and Colucci, D. The HiBall tracker: High-performance wire-area tracking for virtual and augmented environments. In Proceedings of the ACM Symposium on Virtual Reality Software and Technology (Dec 1999).

[144] Ascension Technology website: http://www.ascension-tech.com/

[145] Polhemus Star Trak, http://www.polhemus.com/stardstech.htm

[146] Northern Digital Inc. – ProductsAURORA. http://www.ndigital.com/aurora.html

[147] Raab, F., et al., "Magnetic Position and Orientation Tracking System," *IEEE Trans. Aerospace and Electronic Systems*, Sept. 1979, pp. 709-717.

[148] *Technical Description of DC Magnetic Trackers*, Ascension Technology Corp., Burlington, Vt., 2001. *Personal Comm.*, Oct. 2000, pp. 28-34.

Many other technologies have been used in virtual environments or in support of computer animation.
A CDMA radio ranging approach has been suggested, and many companies sell optical, infrared, and mechanical motion-capture systems. Like MotionStar, these systems are not designed to be scalable for use in large, location-aware applications. Rather, they capture position in one precisely controlled environment.

Some of the limitations of the systems are the following:

1. Steep implementation costs and the need to secure the tracked object to a control unit.
2. Sensors must remain within 1 to 3 meters of the transmitter, and accuracy degrades with the presence of metallic objects in the environment.
3. Electromagnetic sensing is quite expensive and, like IR, range limited, hence unsuitable for large-scale deployment.

System 11: SenSay (Carnegie Mellon)[149]

SenSay is a sensing technology that keeps track of the user's location by using motion sensors (accelerometers). In addition, a GPS device helps to determine the user's position, both outdoors and inside a building.
Its goal is to help the user receive communications in the appropriate way. For example, if the user is in a conference room and is scheduled to have a meeting, the SenSay can send a routine call directly to voicemail. The technology might also appeal to tourists or those concerned with safety due to the real-time positioning of a SenSay.

There are the typical tracking (i.e., GPS) concerns. If the user's location is being gathered and broadcast, then third parties can gather those details as well. For example, students may, for safety purposes, program their phone to let classmates know their whereabouts on campus at night. Once off campus, the students can disable the positioning feature to regain their privacy.

System 12: ParcTab (Xerox)[150]

ParcTab[151] is a Personal Digital Assistant (PDA) that uses an IR-based cellular network for communication. The IR transmissions can be used to determine their locations in the same way as Active Badges are located. The system can be used to implement applications involving context-triggered actions and automatic reconfiguration[152]. ParcTab has also been used to implement a memory prosthesis in which information about the user's context is collected and organized to form a biography (user profiles) that can be consulted at a later time[153].

---

[149] SenSay website: http://www...

[150] ParcTab website: http://www.ubiq.com/parctab

[151] Adams, N., Gold, R., Schilit, B., Tso, M., Want, R. *An Infrared Network for Mobile Computers.* Proceedings of the USENIX Mobile & Location-Independent Computing Symposium, Cambridge Massachusetts, August 2–3 1993. pp. 41–51.

[152] Schilit, B., Adams, N., Want, R. *Context-Aware Computing Applications.* Workshop on Mobile Computing Systems and Applications, Santa Cruz, December 1994.

[153] Lamming, M., Flynn, M. *"Forget-me-not"—Intimate Computing in Support of Human Memory.* Proceedings of FRIEND21, International Symposium on Next Generation Human Interface, Meguro Gajoen, Japan, 1994.

The location service employs a number of tracking subsystems, including an Active Badge server and a UNIX location server. User agents gather the information provided by those subsystems to allow for user-centric operations. This enables the users to have control over location information. They support the more abstract location services that have been constructed: Location Query Service (LQS) and Active Map Service (AMS). While the user agents are responsible for user-centric operations, the LQS provides location-centric services. The LQS is organized by regions, with a Location Broker assigned to each region. The Location Broker maintains a set of references to the located-objects in its area. These references can be anonymous if the user-agent wishes to retain access control. Agents can also chose to delegate their access control to the LQS to improve efficiency.

The AMS provides location information in an abstract hierarchical location model. The service relies on user agents and LQS as follows. Each server contains an active map consisting of a hierarchy of locations with a containment relation (i.e., region-building-floor-room). The area covered by a single server is a region. There is no location service covering more than a region, that is, client must directly access all regions they are interested in. Each server maintains a set of publications (i.e., located-objects, their locations, and other information) and a set of bandwidth-limited subscribers. Both user-centric queries and location-centric queries are supported. The dissemination of subscription updates to multiple clients is performed efficiently using multicast channels.

System 13: SpotOn (Univ. of Washington)[154]

SpotON is an object tagging technology for 3D location sensing based on RF signal strength analysis. Its purpose is to create and analyze a fine-grained indoor location sensing system and the associated services for use within an ubiquitous computing framework. It is built by using RFIDeas badge and AIRID base station. Such an approach combines the advantages of wireless location systems (fine granularity) with that of IR-based systems (detection at a distance).

In general, the SpotON system is similar to RADAR and the 3D-iD system in developing a fine grained tagging technology based on RF signal strength. However, SpotON can archive better resolution and accuracy than RADAR with a much lower cost than the product from 3D-iD. Accuracy and efficiency could be enhanced even further by the addition of sensor fusion techniques such as integrated accelerometers and online building maps.

SpotON tags are useful in presentations, film, or theater production. Interesting information or anecdotes could be "placed" around a site and retrieved dynamically as location aware objects and people travel through a space. For home automation, various consumer electronics and household appliances could present their interface or take action based on the location of people or other tagged objects. Location-sensing technology such as SpotON tags allow multimedia streams to follow roaming users through different media cells.

---

[154] SpotOn website: http://portolano.cs.washington.edu/projects/spoton/

System 14: EasyLiving (Microsoft)[155]

EasyLiving (and Perceptual User Interfaces (PUI)[156]) uses computer vision technology to figure out where things are. EasyLiving attempts to outfit a home environment with stereo vision technology. (The PUI group focuses on enhancing desktop applications with extra input provided by inexpensive camera technology often present on consumer PCs.)

EasyLiving uses the Digiclops real-time 3D cameras to provide stereo-vision positioning capability in a home environment[157]. The PersonTracker uses knowledge of the two cameras' relative locations, fields of view, and heuristics on the movements of people to produce a final report on the locations and identities of people in the room.

Although EasyLiving uses high-performance cameras, vision systems typically use substantial amounts of processing power to analyze frames captured with comparatively low-complexity hardware. State-of-the-art integrated systems[158] demonstrate that multi-modal processing—silhouette, skin color, and face pattern—can significantly enhance accuracy. Vision location systems must, however, constantly struggle to maintain analysis accuracy as scene complexity increases and more occlusive motion occurs.

One of the limitations is the dependence on infrastructure-based processing power, along with public wariness of ubiquitous cameras, can limit the scalability or suitability of vision location systems in many applications.

A person-tracking software continuously updates the measurement which describes the geometric relationship between user1 and user2 and the coordinate frame of the sensor which is observing them. Whatever process is responsible for keeping user1's session available on nearby devices can query EZLGM for all devices that have service areas that intersect with user1's location. The process first looks at types and availability to determine the set of devices which could provide the display, text input, pointing, etc. It then further reduces the list by considering the physical constraints (e.g. visibility) and electronic constraints (e.g. availability), in order to reach a set of usable, available, and physically-appropriate devices. Visibility can be checked by examining all entities along the line of sight between the user and the device and ensuring none have an extent present which represents something that physically blocks user1's view. Then, once user1's location is stable with respect to a set of devices, the session can be directed to automatically move.

The geometric knowledge can be used to assemble a set of UI devices needed for a particular interaction allowing a user the following capabilities:

---

[155] EasyLiving website: http://www.research.microsoft.com/easyliving

[156] Turk, M., "Moving from GUIs to PUIs," *Proc. Fourth Symposium on Intelligent Information Media*, Tokyo, Japan, December 1998. (also Microsoft Research Technical Report #MSR-TR-98-69)

[157] Krumm, J. et al., "Multi-Camera Multi-Person Tracking for Easy Living," *Proc. 3rd IEEE Int'l Workshop Visual Surveillance*, IEEE Press, Piscataway, N.J., 2000, pp. 3-10.

[158] Darrell, T. et al., "Integrated Person Tracking Using Stereo, Color, and Pattern Detection," *Conf. Computer Vision and Pattern Recognition*, IEEE CS Press, Los Alamitos, Calif., 1998, pp. 601-608.

- Physical Parameters for UI's: When a user moves towards another object (i.e., a couch), his/her display is able to follow appropriately because the geometric model provides information which enables the selection of a visible display.
- Simplified Device Control: When a user starts playing the music, it is not necessary for him/her to select particular speakers or other AV components for the task. He/she was able to focus purely on the task, starting music playback, allowing the system to select devices based upon their location.
- Shared Metaphor: When a user turns down the lights, the provided map of the room allows him/her to quickly identify and control the needed devices based on their physical location. Without this representation, he would need to know particular names of the lights, or have some other way of mapping between physical and network identity. The geometric model provides this shared metaphor between the system and the user, allowing a more natural interaction.

System 15: CyberGuide (GeorgiaTech)[159]

Cyberguide measures its position by an IR positioning for the indoor version within a room at a meter resolution, and by GPS for the outdoor version. Because Cyberguide knows its physical location and where it is pointing, it will be able to describe to other Cyberguides where it is and what it is doing, making many cooperative mobile applications possible. The variations of the system, both indoors and outdoors, provide information services to a tourist about his/her current location; for example, the user can find directions, retrieve background information, and leave comments on the interactive map.

System 16: SmartFloor (Georgia Tech)[160]

SmartFloor proximity location system is based on embedded pressure sensors that capture footfalls[161]. The system uses the data for position tracking and pedestrian recognition. It identifies people based on their footstep profiles, which are modeled allowing a system accuracy of about 93%. Since SmartFloor relies on uniqueness of footstep profiles (works well with up to 15 persons) it overcomes many problems of other biometric user identification techniques, like shadows and lightning in face recognition. And, due to the unobtrusive direct physical contact, the system does not require people to carry a device or wear a tag.

Some of the limitations are the poor scalability and high incremental cost because the floor of each building in which SmartFloor is deployed must be physically altered to install the pressure sensor grids.

---

[159] CyberGuide website: http://www.cc.gatech.edu/fce/cyberguide/

[160] SmartFloor website: http://www.cc.gatech.edu/fce/smartfloor/

[161] Orr, R.J., and Abowd, G.D., "The Smart Floor: A Mechanism for Natural User Identification and Tracking," *Proc. 2000 Conf. Human Factors in Computing Systems* (CHI 2000), ACM Press, New York, 2000.

System 17: GUIDE ()[162]

The GUIDE project uses a cellular system based on 802.11. The cells are defined by the range of their WaveLAN 802.11 base stations. Using a wLAN technology deployed in an area leads to medium granularity location information and non-overlapping cells, which causes dead spots where visitors will lose track of their location. The Position Sensor component on a mobile device is able to determine current location via listening for beckons from cell servers since the cells are not overlapped.

System 18: Niblle (UCLA)[163]

The Nibble location system is an indoor location system for mobile devices (i.e. a laptop) equipped with a wireless network card. A device running Nibble can remember a location by simply giving it a name and entering it into the system. Nibble uses the signal quality received from access points (AP) that can be detected at each location and incrementally builds a Bayesian network which can be used to calculate the most likely location for a signal quality signature. The system can be used to remember locations at the granularity of small rooms. It can successfully discriminate between locations roughly 10 feet apart. However, the performance of Nibble is highly dependant on various factors such as the building topology, number of APs, path effects, noise, etc.

Nibble is a stand-alone version of a "fusion service" developed as part of the Multi-use Sensor Environment (MUSE) project[164], which is the system infrastructure to support densely instrumented environments (otherwise known as "smart spaces").

---

[162] GUIDE website: http://www.

[163] Nibble website: http://mmsl.cs.ucla.edu/nibble/

[164] MUSE project: http://mmsl.cs.ucla.edu/muse

APPENDIX A.1

POSITIONING SENSOR TYPES

Wi-Fi

Ekahau's patent-pending location finding technology to enhance the value of their 802.11 wireless solutions. The Ekahau Positioning Engine software can locate a wireless client device, including standard laptops and PDAs, in a Wi-Fi 802.11 network with an accuracy of 1 meter (3.5ft) on average. The accuracy grows with the number of base stations.

The positioning algorithm was originally intended for enhancing GSM-network solution. It is not actually dependent on network technology since it calculates the location from base station signals and compares them to pre-recorded signal patterns and to the map of the location.

Bluetooth

Bluetooth networks provide an underpinning for sharing information among devices located within 3 meters of each other, and could allow multiple devices to share one 802.11b network connection and determine "pico" positioning around a device with known location attributes.
Ultra Wideband (UWB)

UWB is based on radio technology[165]. It is superior to IR tracking, ultrasonic tracking and other radio-based positioning systems in terms of accuracy. Most LPSs use RF based technologies (i.e., Wi-Fi, Bluetooth) can pinpoint objects 10-16 feet. LPS that use UWB (i.e., UbiSense) can pinpoint and track to an accuracy of 6 feet. In addition, UWB's very wide bandwidth allows to avoid the multipath problem, which is common in the indoor world as it reduces the position accuracy. It is speculated by the RF community that UWB will become more common than Wi-Fi over the decade.

IR

IR-based systems provide accurate location information due to short range and line-of-sight, but suffer from several drawbacks: (a) it scales poorly due to the limited range of IR, (b) it incurs significant installation and maintenance costs, and (c) it performs poorly in the presence of direct sunlight, which is likely to be a problem in rooms with windows.

The main incarnation of this location-tracking technology is the Active Badge system (see below) where it consists of a network of fixed IR transmitters/receivers (badge sensors), and a number of mobile IR computers (or 'badges').

---

[165] Ultra Wideband (UWB): http://www.uwb.org

Ultrasound

Among the several technologies for indoor ranging, ultrasonic time-of-sight (TOS) provides the finest granularity, the minimum unit of distance that can be measured accurately[166]. This is the technology of choice for the best known indoor localization systems. Two of the best known systems, the Active Bat and the Cricket system use ultrasound and RF.

However, these systems usually require excessive manual intervention. The Active Bat system requires a priori knowledge of the position of the nodes that compute the location of a node transmitting RF and ultrasound pulses. In addition, triangulation computations to locate the transmitter node are done in a central computer. Hence, this system is centralized, not scalable, and requires manual entry of the receiver node position. In the MIT Cricket system, the location of the beckons has to be entered manually in a database or programmed into the beckons themselves. In addition, this system has specific constraints on the beckons placement to minimize interference between the ultrasound transmissions.

Other Sensor Types

Video

Location information can also be derived from analysis of data such as video images, as in the MIT Smart Rooms project[167]. While vision has unique advantages over other sensors for tracking people, it also presents unique challenges. Such systems, however, have line of sight problem as IR and work well with only a small number of persons in a room with non-frequent occlusions. Accurate object locations can be determined in this way using relatively cheap hardware, but large amounts of computer processing are required. Furthermore, current image analysis techniques can only deal with simple scenes in which extensive features are tracked, making them unsuitable for locating many objects in cluttered indoor environments. Also, a person's appearance in an image varies significantly due to posture, facing direction, distance from the camera, and occlusions. It can be particularly difficult to keep track of multiple people in a room as they move around and occlude each other. Although a variety of algorithms can overcome these difficulties, the final solution must also work fast enough to make the system responsive to the room's occupants.

Electromagnetic and Optical Trackers

Other systems include active/passive electromagnetic and optical trackers. Electromagnetic trackers[168] can determine object locations and orientations to a high accuracy and resolution

---

[166] Hightower, J. and Borriella, G. Location Systems for Ubiquitous Computing, IEEE Computer, vol. 34,8, pg. 57-66, 2001.

[167] Pentland, A. *Machine Understanding of Human Action*. Proceedings of 7th International Forum on Frontier of Telecommunication Technology, Tokyo, Japan, November 1995.

[168] Raab, F., Blood, E., Steiner, T., Jones, H. *Magnetic Position and Orientation Tracking System*. IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-15, No. 5, September 1979.

(around 1mm in position and 0.2° in orientation), but are expensive and require tethers to control units. Furthermore, electromagnetic trackers have a short range (generally only a few meters) and are sensitive to the presence of metallic objects[169]. Optical trackers (for example, see the HiBall system) are very robust, and can achieve levels of accuracy and resolution similar to those of electromagnetic tracking systems. However, they are most useful in well-constrained environments, and tend to be expensive and mechanically complex. Examples of this class of positioning device are a head tracker for augmented reality systems[170], and a laser-scanning system for tracking human body motion[171] (for example, see the Ascension MotionStar system).

Motion Detectors, Pressurized Sensors, and Magnetic Fields

Some systems use motion detectors and reed switches (which monitor movement of people and the positions of doors). Much of this information can be provided using a single, low-powered and untethered device, thus simplifying the physical and computing infrastructure required to support the interactive environment. The Interactive Office[172], for example, gathers information about the activity of the occupants this way.

There is also a large body of existing work in location tracking in support of virtual reality and animation motion capture. Technically, many of these systems can provide valuable insight into developing similar systems for ubiquitous computing. For example, it has been shown that CDMA-like radio technology can be used for precise position tracking (on the order of 2mm grain size) for virtual environments[173]. However, three important issues separate location sensing for invisible computing from most of these systems. First, these systems are often quite expensive and thus not readily deployable in the ubiquitous sense. But more important then cost, many of these systems are not designed to be scalable even to a building wide level – they are designed to capture position well in a single room immersive environment.

Systems using pressurized sensors (for example, see the SmartFloor sysem) identify persons by their footstep force profiles. Though the accuracy of identifying a moving user is around 90%, it is the most unobtrusive way for users to provide their location information to the system. It works, however, only for people, not for other objects such as mobile devices.

Systems using pulsed DC magnetic fields can be used to determine user orientation while another use ultrasound signals to determine user location. While these technologies and systems are very interesting, they generally suffer the same drawbacks as their IR and RF-tag counterparts. Their specialized hardware is generally targeted at niche markets, tending to make the system cost prohibitive, range limited, and unsuitable for large-scale deployment.

---

[169] Ascension Technology Corp. *'Flock of Birds' Technical Description.* Burlington, Vermont, 1994.

[170] Wang, J., Chi, V., Fuchs, H. *A Real-time Optical 3D Tracker for Head-mounted Display Systems.* Computer Graphics, Publication of the ACM SIGGRAPH, Vol. 24, No. 2, 1990. pp. 205–215.

[171] Sorensen, B., Donath, M., Yang, G., Starr, R. *The Minnesota Scanner: A Prototype Sensor forThree-Dimensional Tracking of Moving Body Segments.* IEEE Transactions on Robotics and Automation, Vol. 5, No. 4, August 1989.

[172] Hodges, S., Louie, G. *Towards the Interactive Office.* Proceedings of SIGCHI'94, Boston, April 1994.

[173] Bible, S.R., Zyda, M. and Brutzman, D. (1995). "Using Spread-Spectrum Ranging Techniques for Position Tracking in a Virtual Environment" in the Proceedings of the 1995 Workshop on Networked Realities, Boston, MA, October 26-28, 1995.

GPS

GPS and LORAN[174] are very successful in the wide-area, but are ineffective in buildings because of the reflections of radio signals that occur frequently in indoor environments. In-building radio positioning systems do exist[175], but offer only modest location accuracies of around 50cm or more. Moreover, with GPS pseudolites (a signal generator that transmits GPS-like signals to users in neighborhood), GPS indoor navigation is possible. In 1999, Seoul National University GPS Lab (SNUGL)[176] developed a centimeter-accuracy indoor GPS navigation system using asynchronous pseudolites.

Some server based GPS systems, like SnapTrack, already claim some navigation capabilities indoors. However, such systems are in general only accurate to within a few tens of meters. Although some experimental setups show decent navigation performance, there is a question of whether GPS should be used for such applications in the first place.

---

[174] Sonnenberg, G. *Radar and Electronic Navigation*. Butterworths, 1988.

[175] Feuerstein, M., Pratt, T. *A Local Area Position Location System*. IEE Conference Publication No. 315, 1989. pp. 79–83.

[176] Kee, C., Yun, D., Jun, H., Parkinson, B., Pullen, S., Lagenstein, T. "Centimeter-Accuracy Indoor Navigation Using GPS-Like Pseudolites". GPS World. November 1, 2001

APPENDIX A.2

POSITIONING METHODS

Location system implementations generally use one or more of techniques to locate objects, people, or both. Most of the methods used to define a position are based on geometry computations such as triangulation (by measuring the bearings of an object from fixed points) and trilateration (by measuring the distance). In addition scene analysis, proximity (detecting physical proximity, monitoring wireless cellular access points, and observing automatic ID systems can also be applied[177].

Cell-Based ID

Currently the most widely deployed solution for network-based positioning uses existing data from the network to identify which RD cell site and sector a user is in. As a result, location accuracy is dependent on cell size. Its main advantage is that it requires no new functionality to be added to handsets. This method works for networks such as GSM as well as Wi-Fi as explained earlier.

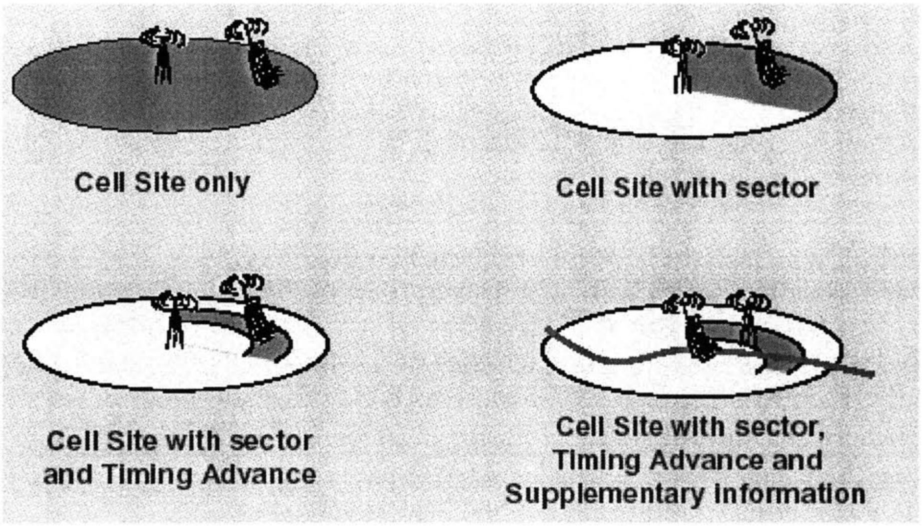


Figure A.2.1: Cell-ID Positioning

Actual cell coverage maps can improve Cell ID positioning center of coverage area will frequently be more accurate that cell site position. In most cases TA does not bring significant improvements in positioning accuracy over Cell ID.

---

[177] A report describing these location techniques in detail can be found at: www.cs.washington.edu/research/portolano/papers/UW-CSE-01-07-01.pdf

Triangulation

Requires at least three base stations that provide signal strength measurements (i.e., from Wi-Fi access points) mapping to an approximate distance. A central server then aggregates the values to triangulate the precise position of the object.

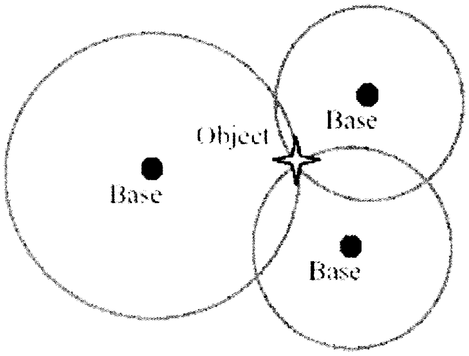


Figure A.2.2: Triangulation

Angle of Arrival (AOA)

Requires at least two distinct signals from known locations of Wi-Fi access points.

Location Fingerprinting

Location fingerprinting is sometimes preferable due to the multi-path environment in indoor areas, where techniques that use triangulation or direction are not very attractive and often can yield highly erroneous results[178] .

Location fingerprinting refers to techniques that match the *fingerprint* of some characteristic of the signal that is location dependent. The fingerprints of different locations are stored in a database and matched to measured fingerprints at the current location of an MS. Some companies have used the multipath characteristics of a signal as its fingerprint. Such techniques require specialized hardware in every base station (BS) (or access point - AP) to correlate the multipath characteristics. In WLANs, an easily available signal characteristic is the received signal strength (RSS). The RSS is a highly variable parameter and issues related to positioning systems based on RSS fingerprinting are not understood very well.

Other research identifies the applicability of current approaches to indoor geolocation in the context of Bluetooth radio signals[179].

---

[178] Pahlavan, K., Krishnamurthy, P. and Beneat, J., "Wideband radio propagation modeling for indoor geolocation applications", *IEEE Comm. Mag*, pp. 60-65, April 1998.

[179] Thapa, K., & Case, S., "An Indoor Positioning Service for Bluetooth Ad Hoc Networks". MICS 2003 Conference. 11-12 April 2003. Duluth, MN.

Mobile IP

Other approaches to detect location change include using the Mobile-IP[180] protocol and network domains.

Mobile IP is the primary example and the current IETF standard for supporting mobility on the Internet. It provides transparent support for host mobility by inserting a level of indirection into the routing architecture. By considering the mobile host's home address as an end-point identifier and not merely as an interface identifier, Mobile IP ensures that the delivery of packets to the host's home address is independent of the host's physical point of attachment. This is achieved by creating an IP tunnel between a mobile host's home network and its care-of address.

Using the Mobile-IP, when the mobile host enters a new zone, it must discover the Foreign Agent (FA) to be assigned a temporary IP address. By installing a context manager service on the same host of FA, the mobile host imports the context of the current zone from the context server just discovered during registration with FA. And, some location-tracking systems are based on network domains, and they are not specifically targeted for indoor or outdoor use (for example, see the GUIDE system). Such a connectivity-based approach to track mobile users can also be realized using Bluetooth.

---

[180] Perkins, C. IP mobility support. RFC 2002, IETF, October 1996.

APPENDIX A.3

LPS SYSTEMS CHARACTERISTICS

Orientation

An important aspect of context (context-aware applications), which is related to physical position, is the orientation of the user holding the mobile device with respect to one or more landmarks in an area. A ubiquitous computing application can benefit from knowing orientation in addition to location for providing the ability to adopt a user interface to the direction in which a user is standing or pointing.

Accuracy

The accuracy of location information needed could vary from one indoor LBS application to another. For example, locating a nearby printer requires fairly coarse-grained location information whereas locating a book in a library would require fine-grained information. The mobile user could expect to get a good result to the request *"print this document on the closest printer"*. Similarly a request like *"show me the map of the area I am in"* will result in the appropriate map being displayed on the screen. Going in the reverse direction, queries such as *"Where is user A?"* would be answered by the system and the location of the mobile user would be returned to within a certain range of his/her actual location.

Some systems (i.e., ActiveBadges, ParcTab) are robust, relatively cheap, and can be integrated into everyday working environments. However, they locate objects only to the granularity of rooms, which act as natural containers for the IR signals emitted by the mobile devices. This limits the extent to which indoor LBS applications can adapt based on information from the LPS. Other sensor technologies give finer-grain location information about objects in the office and home.

The cell size or spacing between grid points influences the granularity of the position estimate. The database entries are collected on a grid of points within the building. Decreasing the spacing (e.g. taking measurements more densely) will increase the database size but are unlikely to yield a better accuracy because the values measured will be more or less the same. On the other hand, if the spacing is very large, it may reduce the search space but drastically decrease the accuracy.

Scale (and Timing)

A LPS may be able to locate objects on different scales – inside a building or within a single room. Further, the number of objects the LPS can locate with a certain amount of positioning infrastructure or over a given time may be limited. To assess the scale of a LPSs the coverage area per unit of the positioning infrastructure and the number of objects the LPS can locate per unit of the positioning infrastructure should be considered. LPSs can often expand to a larger scale by increasing the positioning infrastructure. For example, a LPS based on tags that locates objects in a single building can operate on a campus by outfitting all campus buildings and outdoor areas with the necessary sensor technology.

Also, time reflects an important consideration because of the limited bandwidth available in sensing objects. For example, a RF-based sensor can only tolerate a maximum number of communications before the channel becomes congested. Beyond this threshold, either latency in determining the objects' positions will increase or a loss in accuracy will occur because the system calculates the objects' positions less frequently.

Recognition

For indoor LBS applications that need to recognize or classify located objects to take a specific action based on their location, an automatic identification mechanism is needed. For example, the Buddy Finder trigger-based LBS application presented in **Chapter 2**, uses this mechanism.

Systems with recognition capability may recognize only some feature types. For example, cameras and vision systems can easily distinguish the color or shape of an object but cannot automatically recognize individual people (or other objects).

A general technique for providing recognition capability assigns names or unique IDs to objects the system locates. Once a tag, badge, or label on the object reveals its unique ID, the infrastructure can access an external database to look up the name, type, or other semantic information about the object. It can also combine the unique ID with other contextual information so it can interpret the same object differently under varying circumstances. For example, a person can retrieve the descriptions of objects in a museum in a specified language. The infrastructure can also reverse the unique ID model to emit IDs such as URLs that mobile objects can recognize and use[181].

User Privacy

Establishing private communication is a challenge in location/context-aware applications. Authenticating the supplied location information is difficult because today's sensor systems typically only detect things such as active badges that can be removed from the mobile object they represent.

In addition to protecting the content of the communication, the address of the content should also be protected to prevent leaking of location information. Not surprisingly, most people do not like the idea of being precisely located at anytime, by anyone, especially when the location data is logged. It is important to address privacy issues[182] in context-aware computing.

---

[181] Barton, J. and Kindberg, T. The CoolTown User Experience, tech. report 2001-22, HP Laboratories, Palo Alto, Calif., 2001. a Virtual Environment," Second IEEE Workshop Networked Realities, http://www.npsnet.org/~zyda/pubs/

[182] Privacy international survey: Http://www.privacyinternational.org/survey/technologies.html

Few existing LPS systems, however, provide a satisfactory solution at current stage, while many other systems choose to ignore security and privacy concerns. User should be able to have the control over their contextual information and over who may gain access to it. The system architecture needs to provide user-controllable tradeoffs between privacy guarantees and both functionality and efficiency. But it is difficult to be specific about what context information should be visible to who, and when.

Note, that privacy can also be considered a property of the application, not the underlying architecture. There are detailed guidelines and metrics for preserving various levels of user privacy in applications that ensure that conformance to OECD and ILO privacy guidelines.

If you're going to discuss 'privacy' you need to define what it is. There appears to be a prevailing view in academia that only a peer-to-peer, anonymous architecture can provide privacy. To coin a phrase, that view is not even wrong.

Decentralization

In order to deploy and administer a system in a scalable way it might be necessary to have a decentralized control and management functions. A decentralized system would allow the administrator of a space in a building configure and install a location transmitter that announces the identity of that space. Each transmitter would seamlessly integrate with the rest of the system. Location receiver hardware would be attached to every device of interest to a user. This way, there is no need for a central entity to keep track of each individual component in the system.

Note that a system administrator of a space can use software to configure that space to announce its identity, which implies a decentralized administration. Decentralization is a loaded phrase, and some academics have it that only a system built of autonomous components can 'scale'. However, other systems might support centralized or decentralized administration.

Localized Location Computation (LLC)

Some systems provide a location capability and insist that the object being located actually computes its own position. This model ensures privacy by mandating that no other entity may know where the located object is unless the object specifically takes action to publish that information.

In contrast, some systems require the located object to periodically broadcast, respond with, or otherwise emit telemetry to allow the external infrastructure to locate it. The infrastructure can find objects in its purview without directly involving the objects in the computation. Personal-badge-location systems fit into this category, as do bar codes and the radio frequency identification tags that prevent merchandise theft. Placing the burden on the infrastructure decreases the computational and power demands on the objects being located, which makes many more applications possible due to lower costs and smaller form factors.

The policy for manipulating location data need not be dictated by where the computation is performed.

For example, system-level access control can provide privacy for a movement history in a personal-location system while still allowing the infrastructure to perform the location computation. Doing so, however, imposes a requirement of trust in the access control.

Cost

Costs are associated with time, space, and capital. Time costs include factors such as the installation process's length and the system's administration needs. Space costs involve the amount of installed infrastructure and the hardware's size and form factor. Capital costs include factors such as the price per mobile unit or infrastructure element and the salaries of support personnel.

APPENDIX B

COMMUNICIATON INFRASTRUCTURE

Heterogeneity of Communication Networks

A wide variety of network technologies exist in most building environments (i.e., Ethernet, wireless LANs, cellular, IR, etc). Independent of which technology is used to serve or gain access to information, many services and clients can benefit from learning their location in an automatic way. The communication infrastructure and positioning infrastructure systems may be combined or independent. For example, the Active Badge and ParcTab systems use the same wireless IR link for both tracking and data transfer, and the RF link of the GUIDE and RADAR systems has the same role. Most of other systems reviewed use separate channels.

It might be better to decouple the positioning and communication channels. This separation can utilize the best solution for each problem. The uniform wireless LAN technology, for example IEEE 802.11, has cells with a range of about 100 feet indoors, and this conflicts with a goal of room-level (or better) granularity. Current outdoor wireless MAN/WAN technologies (such as cellular systems) are too coarse for good location sensing.

Network Type

(1) Service-Based

In infrastructure-based systems, like Nexus[183], TEA[184], or the Context Toolkit[185], a specialized context infrastructure serves as a central access point for applications and sensors. These type of networks have nodes that are able to compute their physical location. The nodes compute their location by using ranging technologies (i.e., ultrasound) and location-sensing techniques (i.e., triangulation, proximity). The nodes can thus be used to locate objects. The advantages of a network node knowing its own position, and sharing this information with others, are becoming more and more evident as routing algorithms are becoming smarter and mobile-specific applications are being introduced at the user level[186].

---

[183] Hohl, F., Kubach, U., Leonhardi, A., Rothermel, K. and Schwehm, M.: Next Century Challenges: Nexus - An Open Global Infrastructure for Spatial-Aware Applications. In: Proceedings of the Fifth Annual International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, Washington, USA, 1999

[184] Gellersen, H.-W., Schmidt, A., Beigl, M.: Multi-Sensor Context- Awareness in Mobile Devices and Smart Artifacts. In: Journal on Mobile Networks and Applications, Special Issue on Mobility of Systems, Users, Data and Computing in Mobile Networks and Applications (MONET), Imrich Chlamtac (Ed.), Oct 2002

[185] Dey, A., Abowd, G., Salber, D.: A Context-Based Infrastructure for Smart Environments. In: 1st International Workshop on Managing Interactions in Smart Environments (MANSE '99), Dublin, Ireland, 1999.

[186] K. Pahlavan, Xinrong Li, and Juha-Pekka Makela. Indoor geolocation science and technology. *IEEE Communications Magazine*, 40(2):112–118, Apr 2002.

Applications access the infrastructure to retrieve context information. Sensors are linked to the infrastructure to provide it with their sensor information. Using such an infrastructure allows applications to access context information which has been captured by sensors far away from their current position but requires mobile devices and sensors to be connected to the infrastructure as all communication takes place through the infrastructure. This can become costly, e.g. in terms of energy usage.

In an 802.11b environment, it is natural to exploit access point (AP) signal strengths as perceived by a device to infer location, as no additional hardware or battery power is required to support it. Therefore, many current algorithms for geolocation concentrate on using the current signal-strength or noise signature to discern location. However, the potential cost-effectiveness of using 802.11b signal strengths can come at the expense of accuracy, because 802.11b operates in the 2.5GHz radio band, whose signals are readily attenuated by line-of-site obstructions, and sometimes reflected[187].

Infrastructure-based indoor location positioning research includes systems like Active Badge, ActiveBat, and PinPoint which require the user to wear a transmitter that periodically emits a pulse picked up by a grid of receivers (the infrastructure) whose positions are known and which compute the RF time of flight to determine position of the user.

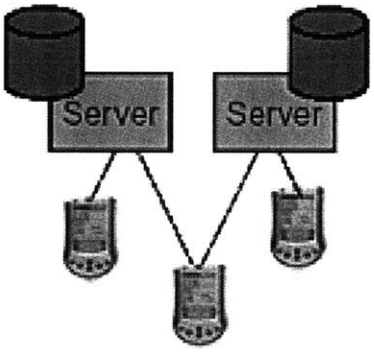


Figure B.1: Infrastructure-Based Network

- Nodes with wireless network connection and location sensor
- Nodes need to obtain their geographic position from the network, which computes their location.
- Location model stored on servers

Context-aware computing relies on an available infrastructure that has a globally accessible data repository. This consistent model information enables distributed applications to interact with their environment according to the model and their location. Mobile nodes in such systems require access to the infrastructure in order to access stored model data.

---

[187] J. Beutel. Geolocation in a picoradio environment. Masters Thesis, Dept. of Electrical Engineering, ETH Zurich, and Dept. of Electrical Engineering and Computer Science, UC Berkeley, 2000.

Location information of the nodes has to be provided to the infrastructure thus they can retrieve location-dependent information. As a result, mobile devices have to obtain their geographic position. This implies mobile nodes with certain capabilities with respect to their communication as well as computing power and sensor inputs (e.g. RFID-reader) for location determination.

An example of such an approach is the NEXUS project where the mobile device on which the application is running has a wireless connection to the infrastructure, e.g 802.11 or GPRS. The mobile device also has some means to determine its current position, e.g. GPS outdoors or an infrared-based system indoors - manual positioning by the user is another possibility. Given such an environment, an application can make use of an infrastructure-based platform such as the Nexus platform that provides context information.

(2) Ad-hoc networks (diffusion-based)

With the development of wireless devices, a large amount of research is being conducted in mobile and wireless communications. This new topic of research focuses in particular on how to use and how to deal simultaneously with all these heterogeneous devices and how to organize them into self configuring networks that do not require a pre-established infrastructure. Such networks are assumed to be formed by mobile devices users carry, e.g. cell phones or PDAs. These devices are equipped with short range wireless communication such as Bluetooth or Wi-Fi in peer-to-peer mode and can obtain their location.

These autonomous networks are called ad-hoc networks. To make these networks autonomous, each node has to collaborate with its neighbours in order to exchange information. Thus nodes can behave at the same time as routers or end systems. Hence, in contrast to the infrastructure-based network, in an ad hoc-based network the context information is retrieved directly from autonomous sensors in the vicinity and stored on the mobile devices.

To obtain context information that is not available locally, mobile devices have to exchange their stored sensor information with other mobile devices. As a result, it is likely that applications will only gain access to context into an infrastructure and the use within an ad hoc-based system.

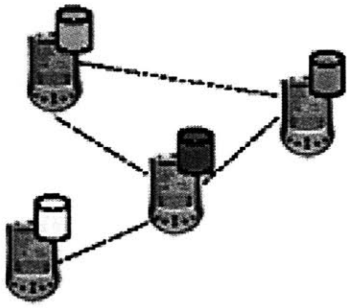


Figure B.2: Ad-hoc Network

- Nodes build ad-hoc network
- Location model stored on devices

Due to the unpredictable mobility of these mobile devices (or nodes) the topology of an ad-hoc network is unstable and likely to change frequently. Routing in such conditions becomes then a challenging task.

The scalability issue in ad hoc networks appears to be an important matter when the number of mobile nodes increases. Indeed IP addresses are still to be used in ad hoc networks in order to maintain utilization of existing applications and to provide with authentication facilities, but routing can not be based anymore on IP addresses network identifiers. As each mobile node joining an ad hoc network is to keep its home IP address in order to maintain communication while moving, then the network ID will not reflect anymore its new network attachment. So mobility issue complicates the routing process as routing tables will have to contain every single IP address and would not be only based on the network part of the address.

Applications using an ad-hoc network do not share the model but rely on their localized location model. Inconsistencies do not occur – at least from the point of view of an application – since decisions based on the model are based on the locally stored data. Ad-hoc networks do not allow the access of services from every position in the network due to partitioning of the network. As a result, applications relying on model data have to store them locally. Different applications on different devices can rely on potentially inconsistent states of the same model.

Relying on locally accessible data helps for optimizing the power consumption. Second, applications on such devices are typically related to the user. Hence, the discovery of services in the proximity of the user can guide him/her through a smart environment and provide a multitude of information.

Mobile ad-hoc networks are constituted through mobile nodes without any a priori known topology, which is highly dynamic. Nodes can be devices include cell phones, PDAs, etc. The interaction of these devices is dependent on the environment. First, long range communication is too costly in either monetary terms or energy consumption.

Sensing object locations with no fixed infrastructure represents a highly scalable and low-cost approach. In the future, infrastructural systems could incorporate ad hoc concepts to increase accuracy or reduce cost. For example, it might be possible for a system like Active Bat to use a sparser ceiling-mounted ultrasound receiver grid if Bats could also accurately measure their distance from other Bats and share this information with the infrastructure.

The SpotON system implements ad-hoc lateration with low-cost tags. SpotON tags use RF signal attenuation to estimate inter-tag distance. They exploit the density of tags and correlation of multiple measurements to improve both accuracy and precision.

The ad hoc-based access to sensors makes context-aware applications possible even on devices that do not have any integrated sensors. The limited communication range of the devices involved ensures that the requesting device is in the vicinity of the node.

In many cases this allows the assumption that the environmental context of the device is similar to the node's context. Additionally, the ad hoc mode of the node can be used to make the context information available to devices that are in its vicinity but too far away to communicate directly. For such situations it has been shown that a flooding-based information dissemination process can effectively distribute information in an ad hoc network[188]. An application that makes use of such mechanisms is the Usenet-on-the-fly[189]. Here information that is grouped according to topics is replicated on devices in the spatial vicinity of the source.

Ad-Hoc Sensor Networking Ad-hoc sensor networking has a large community investigating many issues from distributed computation to cryptography to ad-hoc routing protocols to data dissemination in low power wireless networks. A primary driver for this work is the DARPA SensIT program which seeks to create cheap, pervasive platforms that combine multiple sensor types, embedded processors, positioning ability and wireless communication."[190]

Example: MIT Cricket

MIT Cricket comprises a fixed infrastructure of ultrasound emitters used by mobile nodes to determine their positions by trilateration. There is no centralized authority providing spatial indexing. Instead mobile nodes are assumed to be capable of moderate-bandwidth wireless communication with a backbone network, and to carry sufficient battery power for the cost of communicating to be negligible. Nodes discover and bind to local spatial services in accordance with their own agendas. Services independently choose indexing algorithms as best optimize their application. Unfortunately, the model is best suited to indexing static datasets with the changing positions of enquiring nodes. Determining the positions of and interactions between moving objects requires each to push its location to a central service at regular intervals.

Example: RELATE project

The RELATE project[191] investigates relative positioning in the specific context of tangible interfaces that involve spatial arrangement of physical interaction objects on 2D surfaces, such as white board or tables. Relate is an approach that uses dedicated positioning technology to obtain finer-grained relative position, targeted at close range operation. The research is driven by positioning requirements that we observe in tangible interface systems composed of physical interaction objects. Tangible interfaces have recently attracted considerable research interest, as part of the paradigm shift toward ubiquitous computing, aiming to provide interaction in ways that are intuitive and seamlessly integrated with people's activity in a physical world[192].

(3) Hybrid Networks

---

[188] Ho, C., Obraczka, K., Tsudik, G., and Viswanath, K.: Flooding for Reliable Multicast in Multi-Hop Ah Hoc Networks. In: Workshop on Discrete Algorithms and Methods for Mobility at the Fifth Annual International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, Washington, USA, 1999

[189] Becker, C., Bauer, M., and Hähner, J.: Usenet-on-the-fly: supporting locality of information in spontaneous networking environments. In: CSCW 2002 Workshop on Ad hoc Communications and Collaboration in Ubiquitous Computing Environments, New Orleans, USA, 2002.

[190] DARPA Information Technology Office. Sensor information technology. Website, 2001. http://www.darpa.mil/ito/research/sensit/

[191] RELATE Project http://ubicomp.lancs.ac.uk/relate

[192] P.Dourish :Where the action is: The foundations of embodied interaction MIT Press, 2001.

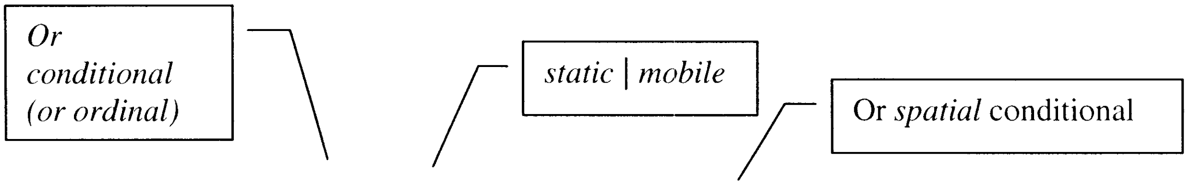
The above approaches describe different application models relying on model data. The infrastructure and ad-hoc approach will coexist in many scenarios. The ad-hoc networks could be used to propagate information that is too short-lived for a centralized storage or only interesting in a small area. The infrastructure-based network could provide the ad-hoc network nodes with model data in distinct areas, thus giving mobile nodes accurate model data for the surrounding area. Therefore, it is feasible to aim at an integration.

Example: Nexus /ContextCube

In an ad hoc environment mobile devices can communicate directly with any ContextCube in their transmission range using a wireless communication interface. Thus, sensors which are also integrated in an infrastructure can be used directly by devices nearby which do not have access to the infrastructure or do not want to use it, because an uplink to the infrastructure may be too costly, either financially or in terms of energy consumption. In such an environment you typically have a very heterogeneous set of devices. Hence, interoperability plays an important role and can be achieved by using an open protocol as provided by AWML and AWQL. Using the same protocol for both, the infrastructure and the ad hoc mode offers the advantage of only having to implement a single interoperability protocol for the ContextCube.

APPENDIX C

EVENT SPECIFICATION LANGUAGE PROTOTYPE



When/If ***<logical>*** ***<object>*** **+ is/are** ***<relationship>*** ***<location>*** **+ when** ***<temporal>*** **then** ***<action>*****+**

Where <location> could be one of the following:

- in (true symbolic relationship or static state)..could be also within (a wi-fi cell)
- alone in (true symbolic relationship or static state)
- not in (true symbolic relationship or static state)

Where <relationship> could be one of the following

- entering
- leaving

The following table shows how English prepositions can be broken down into logical, spatial, and temporal components of location (or context).

Preposition	***<logical>***	***<object>***	***<relationship>***	***<location>***	***<temporal>***
About	about a dozen			about the room	about noon
Above	numbers above 6			above the table	(not temporal)
Across	(not ordinal)			across the table	across the years
After	after John alphabetically		After the table?	(not spatial)	after July
Against	(not ordinal)			against the table	(not temporal)
Ahead	ahead of the others			(not spatial)	ahead of schedule
Alone			Alone in	("in" is spatial)	
Along	(not ordinal)			along the river	(not temporal)
Among	among the attendees		(not temporal)	?	
Amongst	(not ordinal)			(not spatial?)	(not temporal)
…					
In (not in)			Entering/Leaving	Inside the room	

Content comparison and equivalence

These following operators can be used to compare events of the same class, or compare specific event fields against specified values:
=, <, >, !=, <=, =>

For example, the following returns a set of events whose <field1> and <field2> both compare to the respective values x and y as specified:

((a)<field1> <operator> x and <field2> <operator> y)

The following is a set of events (type a or a combination of type a and type b) whose either <field1> or <field2> (as applicable) compare to the respective parameters as specified. These types do not have to denote the same event class, nor to have the same set of attribute fields.

((a) <field1> <operator> x or (b) <field2> <operator> y)

Temporal context

((<event>|<interval>) before (<event>|<interval>) [without <event>] [within <timevalue>])

The following returns the set of all a which are followed at some point by type a b, where type a and type b can be either events or intervals. (type a before type b within <timevalue>) returns the set of all a that are followed by type a b within the specified time period denoted by <timevalue>. (type a before type without type c) returns the set of all a that are followed by type a b and no type c occurred in between.

Similar to the previous construct, but in this case the type a is preceded at some point by type a b are considered.

((<event>|<interval>) after (<event>|<interval>) [without <event>] [within <timevalue>])

In the following, (type a equal type b) returns the set of all a that occurred at the same time as type a b.

(<event> equal <event>)

Interval operators

An interval is defined as having a start time and an end time, and can encapsulate any number of events occurring within those inclusive boundaries. When an interval is defined within a query, a new object instance of the interval class type is constructed, and this provides the following methods that can be employed within the query itself. Some maintenance functions are provided like; start(), which returns the time value denoting the start of the interval, end(), the end boundary, events(), the number of events within it, and eventAt(x), the event at position x.

An interval is specified as follows:
(: (<timevalue> | <event>) (((until | to) (<timevalue> | <event>)) |for <timeamount>) :)

"(: and :)" denote that an interval is being specified and to differentiate between excluding or including the second time value itself. These operators default to a 'start-point' consumption model. A start time is matched with the closest end point and then not used again in any further matches. If for is used the second parameter must be a relative amount of time, which is then added to the starting boundary to compute the ending boundary of the interval.

There are a number of relationships between intervals, all of which can be specified in terms of the start() and end() operators of the interval object class. before and after have already been discussed in the context of event operators.The other operators we have provided reflect the ways in which an ordered pair of intervals can be related. These are equal, meets, overlaps, during, starts, finishes, contains and their inverses. Intersect and join are interval operators that create new intervals on which queries can be applied. At the end, see the OGC Filter Encoding Specification's Filters, which can be adopted for such as event language. This specification already has the 'contains,' 'intersects,' 'joins,' and 'overlaps' elements.

Composite/sequence pattern operators

A composite event is denoted as an event defined in term of a sequence of other basic or composite events. It exists only within the context of the current query and is defined by specifying a path template, also equivalent in meaning to a sequence or pattern. Operations on paths are carried out by enclosing the template in [: :], as in
[: A followed by (B or C) without E :]

A composite 'path' does not return single event instances, but rather constructs a set of new composite event objects (if successful) which each contain pointers to all the basic events they were made up from within the session timeline. Functions are provided which can extract relevant information from the resulting construct.

Syntax Examples

Consider the following event: *"when Cliff entered room 101 and Kris was there."*

One way of expressing this is to find the period when Kris was in *101* and then see if Cliff was seen within that period in the same room.

fromwhere (badge.Name/Id=' Cliff.K and badge.location='R101)
during (: (badge.Name/Id='Kris.K and badge.Location='R101)
until (badge.Name/Id=' Kris.K and badge.Location!=' R101) :)

Or "when Kris walked from the office-room to his office (R101)) and started editing a document."

This will return all event sequences which match the specification provided. A simple way of specifying this is:

eventsOf([:(badge.Name='Kris.K and badge.Location='Office-Room')
followedby (badge.Name=' Kris.K and badge.Location='R101)
followedby (workstation.documentedit.user=' Kris.K):]
during (: 10:00:AM to 11:00:AM :))

To summarize, it is possible to find events relative to other events that have occurred before or after them, as well as with respect to specified bounded intervals. Intervals can be operated on to determine overlapping periods, and if desired, output can be filtered so that only specific event types, or events with specific parameters, are returned.

	Preposition	**Temporal example**	**Spatial Example**	**Ordinal Example**
1.	About	about noon	about the room	about a dozen
2.	Above	(not temporal)	above the table	numbers above 6
3.	Across	across the years	across the table	(not ordinal)
4.	After	after July	(not spatial)	after Joh alphabetically
5.	Against	(not temporal)	against the table	(not ordinal)
6.	Ahead (of)	ahead of schedule	(not spatial)	ahead of the others
7.	Along	(not temporal)	along the river	(not ordinal)
8.	Among	(not temporal)	among the daisies	among the leaders
9.	Amongst	(not temporal)	(not spatial?)	(not ordinal)
10.	Around	around 1776	around the table	around a hundred
11.	As			
12.	At	at noon	at the table	at the halfway point
13.	Before	before 6 PM	before the fireplace	I was served before you
14.	Behind	behind schedule	behind the table	behind John in line
15.	Below	(not temporal)	below the table	below average
16.	Beneath	(not temporal)	beneath the table	(not ordinal)
17.	Besides			
18.	Beside	(not temporal)	beside the table	(not ordinal)
19.	Between	between 6 &7 o'clock	between table & lamp	between 40 and 49
20.	By	by Saturday	by the table	(not ordinal)
21.	Down	(not temporal)	down the hill	(not ordinal)
22.	During	during June and July	(not spatial)	(not ordinal)
23.	For	for an hour	(not spatial)	(not ordinal)
24.	From	from 6 to 7 o'clock	from the table	from 95 to 97 octane
25.	In	in the Pennsylvanian	in the room	in the top 50
26.	Inside	inside of an hour	inside the cupboard	(not ordinal)
27.	Like			
28.	Near	near Christmas	near the table	near par
29.	Nearby	(not temporal)	nearby the table	(not ordinal)
30.	Next, next to	(not temporal)	next to the table	next in line
31.	Of	(possessive only)		

32. Off	(not temporal)	off the table	off the scale
33. On	on Thursdays	on the table	(not ordinal)
34. Onto	(not temporal)	onto the table	(not ordinal)
35. Out	(not temporal)	out of the box	out of order
36. Outside	(not temporal)	outside the lines	(not ordinal)
37. Over	over an hour	over the table	over 60 mph
38. Till, until	until midnight	(not spatial)	(not ordinal)
39. Through	through the week	through the field	(not ordinal)
40. To	Mon. to Fri.	to the table; to the left	cooled to 50 deg.
41. Toward	toward midnight	toward the table	(not ordinal)
42. Under	under an hour	under the table	under 100 pounds
43. Until	until Saturday	(not spatial)	until age 7
44. Up	time is up	up the wall	up the scale
45. Upon	once upon a time	upon the table	(not ordinal)
46. While	while the TV is on	(not spatial)	(not ordinal)
47. Whilst	(not temporal)	(not spatial)	(not ordinal)
48. With	(not temporal)	(not spatial)	(not ordinal)
49. Within	within an hour	within the room	within 3 degrees
50. Without	(not temporal)	(not spatial)	(not ordinal)

OGC Filter Encoding Specification's Filter Expressions

Filter: defines filer expressions	
Element Name	**Description**
-- Spatial Operators <spatialOps> –	
Equals	Defined in section 3.2.19.2 of the OpenGIS Simple Feature Specification for SQL
Disjoin	Same as above.
Touches	Same as above.
Within	Same as above.
Overlaps	Evaluates whether the value of the specified geometric property and the specified literal geometric value (expressed using GML) spatially overlap. Defined in section 3.2.19.2 of the OpenGIS Simple Feature Specification for SQL.
Crosses	Defined in section 3.2.19.2 of the OpenGIS Simple Feature Specification for SQL
Intersects	Same as above.
Contains	Same as above.
DWithin	Test whether the value of a geometric property is within or beyond a specified distance of the specified literal geometric value. Distance values are expressed using the <Distance> element. The content of the <Distance> element represents the magnitude of the distance and the units attribute is used to specify the units of measure. The units attribute is of type anyURI so that it may be used to reference a units dictionary.
Beyond	Same as above.
BBOX	Encodes the bounding box constrained based on the gml:Box geometry. It is equivalent to the spatial operation <Not><Disjoint> … </Disjoint></Not> meaning that the <BBOX> operator should identify all geometries that spatially interact with the box in some manner.
-- Comparison Operators <comparisonOps> – Note: For our purpose, "Property" can mean "Object" or "User"	
PropertyIsEqualTo	Defined in the OGC Common Catalog Query Language.
PropertyIsNotEqualTo	Defined in the OGC Common Catalog Query Language.
PropertyIsLessThan	Defined in the OGC Common Catalog Query Language.
PropertyIsGreaterThan	Defined in the OGC Common Catalog Query Language.
PropertyIsLessThanOrEqualTo	Defined in the OGC Common Catalog Query Language.
PropertyIsGreaterThanOrEqualTo	Defined in the OGC Common Catalog Query Language.
PropertyIsLike	Encode a character string comparison operator with pattern matching.
PropertyIsNull	Encodes an operator that checks to see if the value of its content is NULL. A NULL is equivalent to no value present. The value 0 is a valid value and is not considered NULL.
PropertyIsBetween	Defined as a compact way of encoding a range check. The lower and upper boundary values are inclusive.

-- Logical Operators <logicalOps> –	
AND	Used to combine scalar, spatial and other logical expressions to form more complex compound expressions.
OR	Same as above.
NOT	Same as above.
-- Arithmetic Operators <arithmeticOps> –	
Add	Encodes the operation of addition and contains the arguments which can be any expression that validates according to the OGC Filter Encoding specification.
Sub	Encodes the operation of subtraction where the second argument is subtracted from the first. The <Sub> element contains the argument when can be any expression that validates according to the OGC Filter Encoding specification
Mul	Encodes the operation multiplication. The <Mul> element contains the two argument to be multiplied which can be any expression that validates according to the OGC Filter Encoding specification
Div	Encodes the operation of division where the first argument is divided by the second argument. The <Div> element contains the arguments which can be any valid expression that validates according to the OGC Filter Encoding specification. The second argument or expression cannot evaluate to zero.