Enhancing the Location-Context Through Inference Over Positioning Data

Filipe Meneses, Adriano Moreira

Department of Information Systems, University of Minho, Portugal
{meneses, adriano}@dsi.uminho.pt

Abstract. The current ubiquity of wireless communication networks is enabling new ways to compute the position of mobile devices and their users. One good example is the Place Lab system that exploits the beacons broadcast by many wireless networks to estimate the geographic position of mobile users. Although position is a fundamental dimension in the user's context, other location-related dimensions of the context are also important for the development of innovative context-aware applications. In this paper we propose a new architecture for context acquisition and management, as a new layer added over the Place Lab system, where inference techniques over the basic positioning data are used to enrich the context with new dimensions without compromising user privacy.

1 Introduction

The availability of information about the position and location of mobile users is fundamental to the creation of location-based services and applications. Today, the position of users can be acquired using a wide variety of devices or technologies, making it available locally in the user device or as a network service. However, the majority of these solutions only work in specific conditions (with certain equipment, where location is acquired through a specific technique or device, etc.), or raise significant privacy issues. Therefore, the big challenge in this field is to provide the position of users in a way that enables the easy development of location-aware applications while preserving the privacy of their users.

The Global Positioning System (GPS) is one of the most well known and popular positioning technologies, but has the disadvantage of working only in open areas, in line-of-sight with the satellites. A number of other specific devices were also developed to allow the acquisition of the user position inside buildings [1,2]. However, all these technologies demand the existence of a specific network of sensors.

In the last few years, WiFi networks have been installed in an increasing number of places, and are available within many buildings and in some public spaces like shoppings malls, hotels, public squares, etc. For these networks, the basic positioning mechanism is based on the identification of the BSSID of the current cell, positioning the user's device as being inside the coverage area of the corresponding access point. More precise results are achieved in systems where the network is previously

PREPRINT
Proceedings of the Conference on Mobile and Ubiquitous Systems, Guimarães, Portugal, 29-30 June, 2006

calibrated and the positioning calculation is done by triangulation of the signals received from several access points.

GSM and other cellular networks cover most of the territory of the countries where these networks were adopted and, due to the demand for more network capacity, the number of cells in these networks is rapidly increasing while their size is becoming smaller. Besides providing voice and data communications anywhere, these networks can also be used as an infrastructure to find the position of its users. TOA (Time Of Arrival), TDOA (Time Difference of Arrival), Enhanced Observed Time Difference (E-OTD) and Angle of Arrival (AOA) are positioning techniques that allow network operators to locate the mobile devices inside the network coverage area. However, these positioning techniques make the position available to the network operators, who rarely allow individual users to access their positioning data. A more simple positioning technique for these networks is to locate the user's device by identifying the cell that is being used. Although it does not produce very accurate results (especially in rural areas where a cell can cover several square kilometres), this cell positioning can be done on the user's device, without depending on any network service.

Although there are a number of different technologies to acquire the user position, the spread of location-based services and applications is not yet a reality. To bootstrap the creation of location-enhanced applications, the Place Lab project [3] created a framework with the intent to enable the easy development of location-based services and applications. The basic idea behind the Place Lab framework is that the widespread wireless communications infrastructures, and the radio beacons they broadcast, can be exploited by mobile devices to estimate their own position without compromising their privacy. The positioning principle also calls for a cooperative effort between users to build maps of geo-referenced beacons that are exchanged among all the users, and used for the calculation of the position.

In addition to a positioning technique, the Place Lab system runs on a multitude of computing platforms and provides a few interfaces from where applications can access the position data, thus making it a very general and flexible solution.

By adopting the Place Lab framework, developers can access the position of mobile devices, expressed by a pair of geographic coordinates, by querying a service that runs in the user device itself. Developers use the service provided by the Place Lab infrastructure, without having to deal with any detail concerned to the positioning acquisition. Place Lab supports positioning through WiFi networks, GPS, Bluetooth beacons and GSM cell-ID, and runs on mobile phones, PDAs and laptop computers. Position, however, is only one among many other parameters that can be used to describe the context of mobile users.

The context of a user is everything that characterizes his environment: where he is, with whom he is or even how familiar he his with the surrounding environment. Place Lab provides a basic form of context by providing the position of a device as a pair of geographic coordinates. In this paper we propose an extension to the Place Lab architecture that will enable richer forms of context. We do not change the Place Lab interfaces, we do not add any system to acquire the user location through the use of other positioning technologies, and neither do we compromise the privacy of the user data. Instead, we add a new layer on top of the Place Lab infrastructure, based on a set of modules that use the Place Lab basic positioning data to produce new context dimensions through calculation, lookup and inference.

In the next section we describe other initiatives that are related to the work described in this paper. In section 3 we present the basics of the Place Lab architecture and, in section 4, we describe the new layer added to the framework. In the fifth section we describe the validation plan proposed to assess the validity of the proposed approach and, in the last section, we present our conclusions.

2 Related Work

The concept of an architecture/framework to support the development of context-aware applications for mobile devices has already been addressed by other authors.

ContextPhone [4] was designed to fill the gap between the mobile phone operating systems functionality and the features that applications developers need. It is composed of four interconnected modules: the sensors to acquire the context data from different sources, the communications module to connect to external services, some customizable applications and a set of system services which include logging and background services. ContextPhone presents a context definition based on location (acquired through GPS and GSM cell-ID) and in the data retrieved from the continuous monitoring of a phone's internal state and usage (including the user interaction, communication behaviour and physical environment).

Mobile Bristol [5] provides a real-world test bed for research in pervasive computing. The project goal is to provide a rapid and convenient development path for researchers, creatives and others who are interested in exploring new values for context-sensitive applications and services but may not necessarily have programming skills. The idea behind Mobile Bristol is to provide an experimental vehicle that facilitates research in pervasive computing. In Mobile Bristol, context is a specification of behaviour written in the context language: it is a set of event handlers and resources that collectively define the response of a client to its user and environment.

Context Toolkit [6] also intends to facilitate the development of context-aware applications. It has a set of software components that provide applications with access to context information through a network API. The context widgets insulate applications from context acquisition concerns.

Although several frameworks exist, the solutions scopes are very different. Mobile Bristol is not a generic solution because it was deployed only in Bristol city centre, it demands network connectivity and presents a context that is a specification of applications behaviour and not a representation of the user surrounding environment. Similarly, Context Toolkit is also dependent on a set of infrastructure devices and is network based. Although ContextPhone provides a broad notion of context and provides a set of services that may be useful to advanced developers, this solution is limited to Symbian OS phones and the addition of some of the new functionalities requires a recompilation of the core software.

Place Lab joins in one platform all the advantages present on the other platforms but limits the context to the geographic position. It runs on different devices and provides the same service in all of them, it ensures privacy and it is not dependent on any network services. By expanding the context it becomes a truly competitive platform that may achieve the goal of bootstrapping the dissemination of location-based services and applications.

3 The Place Lab Architecture

It has passed more than one decade since researchers started to create and deploy location-based services. However, until today, no one has achieved a commercial success due to technical constrains.

In [7], the creation of the Place Lab platform is justified by the need of breaking a perverse cycle: there are few users of location-enhanced applications because there are few of these applications and the number of applications is small because there are few developers inventing location-enhanced applications as a result of a small target community. The target community is small because there is no common platform for location-enhanced applications, because the providers do not invest in new location-capable hardware which ends up in few users of location-enhanced applications.

Place Lab architecture has three layers (figure 1): the *spotters* used to acquire positioning; the *mapper* layer that converts the raw data acquired by one of the spotters in a geographic point and; the *tracker* layer that solves conflicts that may exist in the positioning data acquired by different spotters and provides an interface that allows applications to request the current position of the device.

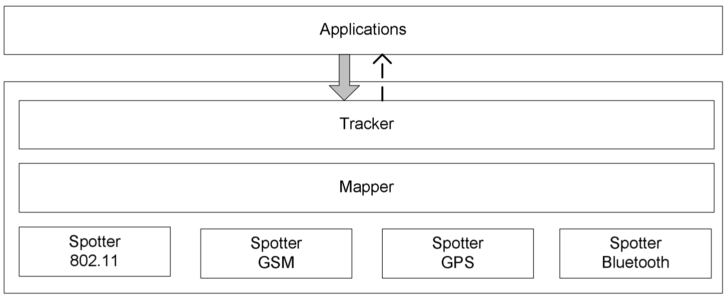

Fig. 1. The Place Lab architecture

3.1 Acquire Positioning Data

To achieve the goal of bootstrapping the creation of location-aware applications and services, Place Lab supports positioning through several mechanisms.

Although the positioning data can be acquired through very different technologies, Place Lab only uses those that are more commonly available and those which preserve the user privacy. It does not use any network service because it could compromise the user privacy and would be dependent on the availability of the service.

On the first level of Place Lab architecture are the *spotters* for each of the supported positioning mechanism:

- GPS – The position can be obtained using a GPS receiver, acquired as a pair of geographic coordinates;
- IEEE 802.11 – Retrieves the BSSID of the neighbour IEEE 802.11 cells;
- Bluetooth – This spotter scans for Bluetooth devices and provides the position data as “being in the neighbourhood” of the sensed device;
- GSM – Retrieves the cell-ID of the current GSM network cell.

3.2 Raw Data Converted into Geographic Positions

Bluetooth, GSM and WiFi spotters give the device position in a symbolic referential. To know that the device is in a certain WiFi network or within a specific GSM cell does not provide very useful information unless a network map is available with the geographic position of each cell. Otherwise, the position information must be used by an application or service that was specifically developed to use symbolic data.

The *mapper* layer provides the transformation of the raw data acquired by the spotters into geographic coordinates. This is done by retrieving the geographic position of the beacons’ sources (WiFi access points, GSM base station, etc.) from a previously built database. In the Place Lab concept, this database is cooperatively built by the users, by geo-referencing beacon sources while using a GPS receiver, and by sharing the collected data with other users through a central network service. In what concerns access to this central service, privacy is ensured by several mechanisms: the user device can retrieve data for the current location anonymously, can pre-fetch data for several possible locations, or it can request data for several very distinct locations, blocking any possible attempt to track the user movements.

3.3 Solving Data Conflicts

The tracker layer manages the position data, solving conflicts that may result from the use of several spotters to acquire the device position and provides an interface for the applications to request the device position.

Because positioning data can be acquired through several spotters simultaneously, conflicts may rise from disjoined positioning data retrieved from different spotters. The tracker layer solves the conflicts and provides the position as a geographic point. The current position of the device can be accessed by applications through one of the five different interfaces provided by the tracker layer, including a local HTTP interface.

4 Adding New Context Dimensions

With the Place Lab layered platform, applications developers use the tracker interface to request the user position without having to deal with any technical detail about how the positioning data is being acquired. Being able to access the position everywhere without demanding additional hardware or relying on a network service is a major advantage over previous known solutions. However, Place Lab only gives the position as a pair of geographic coordinates.

Context is more than location. User context is all that may be used to characterize the surrounding environment: it is the geographic coordinates but also information about if the user is at home, at work, with friends, in an unknown place, etc. Dey [8] defines context as everything that characterizes a user, from location to any other data that characterizes the user and the user environment (like the presence of friends, the use of certain devices, the user's activity, etc).

A high number of applications and services can benefit from the existence of other elements in the context besides the geographic position. As an example consider a guidance application that uses geographic position to select the map to be shown to the user. When the user travels in an unknown place he will appreciate the help he receives from the application in order to arrive to a destination without getting lost. However, the same application should not provide any help at all when the user is in the vicinity of his house because he knows the area perfectly well and, for sure, he does not need any computerized help (especially if the guidance system does not use any traffic information service to select the route with less traffic or uses outdated maps). In this case, the guidance system should use the geographic position to retrieve the appropriate map for an area, but also consider the degree of familiarity the user has with the surrounding area.

By adding new modules to the Place Lab architecture it is possible to improve the description of the user's context, by estimating and calculating new context dimensions, so improving the context whilst still ensuring the context data is kept private.

Figure 2 shows the proposed new architecture, built on top of the Place Lab system, and illustrates where new modules are responsible to compute a more complete context description and to provide an interface for the applications to access it. The central concept of this new layer is that the basic positioning data (symbolic or geographic) can be used to feed higher level context inference modules. The internal architecture of the proposed new layer is built around a flexible and scalable set of modules where a variable number of "learners" and "recognizers" exploit the basic data to produce higher level dimensions of the user context.

The system is composed of a *logger* that collects data from the Place Lab system, a set of *learners* that create new knowledge (by computing the data collected by the logger), a set of *recognizers* that update the user context, and a web server that provides an HTTP interface where the client applications can access the new context data dimensions. The learners are controlled by a scheduler that determines the execution cycle of the learners. The context object holds persistently the context data and the knowledge generated by the learners.

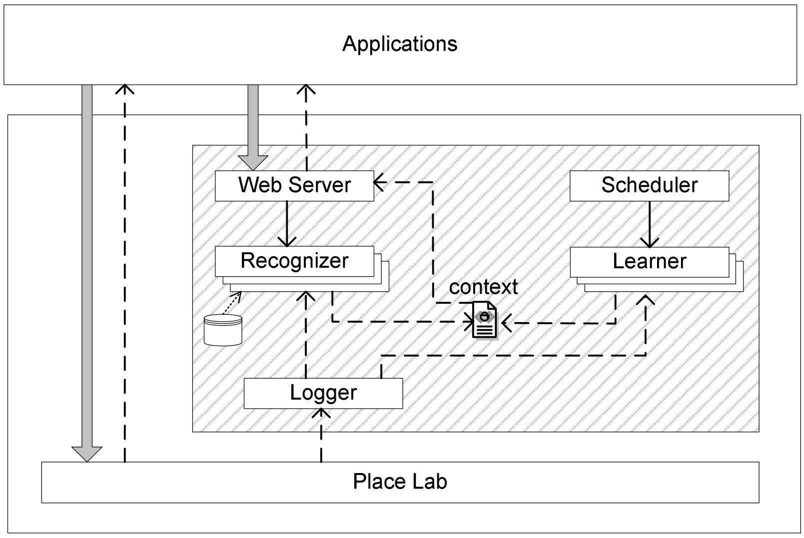

Fig. 2. Place Lab architecture with new software modules

4.1 Acquire Basic Data

The logger module implements the connection between the Place Lab architecture and the new layer. In Place Lab, applications access the position of the device through the tracker interface. In the new layer, the logger module acts has an application in the Place Lab architecture and uses the tracker interface to request the same data that any application can access (the estimated position) and, additionally, it queries directly the GSM and 802.11 spotters to access the cell-ID and BSSID information.

The role of the logger is to collect data periodically and to keep those records in memory for future use by the learners and recognizers. It includes an interface through which the learners and recognizers can retrieve the beacons eared by the device and the geographic position estimated by the Place Lab tracker.

4.2 Create New Context Data

Data collected by the logger will periodically be used by the learners modules to create new contextual data. A learner is a software module capable of creating new context data through the computation of the data collected by the logger. Different learners will produce different contextual data by processing all or just some of the logger records. In our prototype we included two different learners:

– GSM Familiarity Learner – using only the data collected by the logger from the GSM spotter (the cell-ID records) this learner creates a list of symbolic places

visited by the user and estimates the level of the familiarity the user has with each of these places. This is a generic learner that works in any GSM network, without any previous knowledge about the network topology. It tracks the user movement by computing a mobility index based on the time spent on each cell during a given period of time, and considering the set of different cell-IDs collected in that same time period [9]. The places recognized by this learner are not identified by an address or pair of geographic coordinates but by a fingerprint created from the set of cells observed during the time the user stayed in that place.
– Home/work learner – this learner uses the geographic position data to automatically learn the user's home location and the user's work location [10].

Other learners may be developed in order to enrich even more the context. It may include learners that use simultaneously more than one of the basic types of data collected by the logger or even learners that use the previous results of others learners.

A learner can use a network-based service, provided by a third party, to enrich the user context. However, besides the communications costs, these learners raise privacy concern issues. The user may not trust the third party service provider, considering that it may track their movements or make their location data public.

4.3 Context Update

A learner is a process that is executed periodically and every time it runs it uses all the new data collected by the logger since the last execution cycle. However, some learners may be run infrequently. For example: the GSM learner uses the cell-ID records to learn the places known by the user and to compute the familiarity index. Because familiarity index changes slowly, this learning process can be executed only once every fifteen minutes or more.

Some learners may take several seconds to process all the data available in the logger. The amount of processing time varies from learner to learner but also depends on the number of records to be processed and from the processing capabilities of the device.

When a client application requests the user context it must be up to date. Thus, it is necessary to process the last records collected by the logger in order to update the context before it is used to reply to the client application request. A *recognizer* is a software module capable of transforming the data stored in the context module and the most recent records collected by the logger into a context description.

In our first prototype we created three different recognizers that process the most recent records using the knowledge generated by the learners:

– Recognizer for GSM learner – it uses the most recent cell-ID records and the knowledge generated by the GSM learner to estimate the familiarity index of the user with the current location [9];
– Recognizer for home/work learner – It uses the most recent geographic coordinates and the knowledge generated by the Home/work learner to estimate the probability of the user be at home or at work;

– Address recognizer – it uses the current position (estimated geographic coordinates) to query a local database for the corresponding address, enriching the user context with the address that corresponds to the location of the user. Because the storage capabilities of a device like a mobile phone are small, the local database only provides address information for a small geographic area. New data for other regions should be retrieved from a central service, in a way similar to that used to retrieve beacon maps.

To each learner there is a corresponding recognizer but other recognizers may exist as long as they are capable of updating the user context in a very short amount of time. The "Address recognizer" is an example of a recognizer that is not associated to any learner but contributes to the user context.

Our application runs in daemon mode and client applications can use the HTTP interface provided by the embedded web server to request the context description. This HTTP interface extends the Place Lab interface by adding the capability to provide richer contextual information to applications. When an application request is received, the recognizers are executed to update the context that is returned to applications as an xml stream.

In a short period of time, several requests can be received from different client applications. Although the user context is something that can change quickly, the context dimensions present in our system are not likely to change fast enough to justify the execution of the recognizers for each context request received. Thereby, the user's context is considered as up to date for a certain period of time (currently 30 seconds). The *context* module acts as a persistent storage area for the system, holding the user context updated by the recognizers and also the user history and the results of the learning processes produced by learners. Learners process data and create knowledge that is stored persistently in the context object, thus ensuring that all the generated knowledge is retained even if the system is turned off. This is a significant difference to Place Lab that does not have memory - only the current estimated position is available.

5 Evaluation

The GSM and geographic learners were previously developed and tested. A preliminary test was conducted by two users during several weeks and the results were achieved by processing off-line the data previously collected. The analysis of the results shows that the GSM Familiarity Learner was able to correctly identify the places visited by the user in 75,8% of the occasions. The results also show that the familiarity index estimated for each place grows as the user spends more time on the places, reaching a value of approximately 70% for the person's home place with just 6 weeks of data. Further details about the GSM and the geographic learners are described in [9] and [10].

In order to evaluate the entire architecture we are building an application that will assess the quality of the produced context data by querying real users in real time. This application will be executed on Symbian mobile phones because they are widely used by people everywhere, they are turned-on more time than a laptop computer and are kept turned-on while people move from one place to another (which makes the context change gradually, reflecting the user movement). The goal of the application is to assess if the learners and corresponding recognizers produce correct information and if other recognizers (like the recognizer that uses a space model server or a local database with addresses) also produce correct information. The application will contain a set of rules that will define when the user should be queried about the data correctness and will create a log with the answers for later statistical analysis.

6 Conclusion

Having the user context acquired easily at the same time that privacy is ensured is fundamental to bootstrap the dissemination of location-based services and applications. In this paper we presented a model that uses the basic data provided by a generic platform and improve it, generating a richer context with new dimensions that will enable a new set of context-ware applications and services. Context is no longer a geographic point but a set of dimensions that better characterize the user environment. Through a future evaluation of the proposed solution we expect to show the quality and usefulness of the new context dimensions for the development of location-aware mobile applications.

References

1. Want, R., Hopper, A., Falcão, V., Gibbons, J., The active badge location system. ACM Transactions on Information Systems, Vol 10(1), (1992) 91-102
2. Ward, A., Jones, A., Hopper, A., A new location technique for the active office. IEEE Personal Communications, Vol 4(5), (1997) 42-47
3. Place Lab web site, 2006, http://www.placelab.org
4. Raento, M., Oulasvirta, A., Petit, R., Toivonen, H., ContextPhone: A Prototyping Platform for Context-Aware Mobile Applications. IEEE Pervasive Computing, Vol 4(2) (2005) 51-59
5. Mobile Bristol web site, 2006, http://www.mobilebristol.com/
6. Salber, D., Dey, A., Abowd, G., The Context Toolkit: Aiding the Development of Context-Enabled Applications. Conference on Human Factors in Computing Systems (CHI '99). Pittsburgh, PA, USA (1999) 434-441
7. Schilit, B., et al. Challenge: Ubiquitous Location-Aware Computing and the "Place Lab" Initiative. 1st ACM international workshop on Wireless mobile applications and services on WLAN hotspots (WMASH2003). San Diego, CA, USA (2003) 29-35
8. Dey, A., Abowd G., Towards a Better Understanding of Context and Context-Awareness. Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing. Lecture Notes In Computer Science, Vol. 1707. Springer-Verlag (1999) 304-307
9. Meneses, Filipe, Moreira, Adriano, Using GSM CellID Positioning for Place Discovering, submitted to the Tenth International Symposium on Wearable Computers (ISWC'06), Montreux, Switzerland, October 2006.
10. Moreira, Adriano, Santos, Maribel, From GPS tracks to context - Inference of high-level context information through spatial clustering, in II International Conference & Exhibition on Geographic Information - GIS Planet 2005. Estoril, Portugal.