

An Architecture for Location Aware Applications*

James Nord, Kåre Synnes, Peter Parnes
 Department of Computer Science
 Luleå University of Technology, 971 87 Luleå, Sweden
 +46 920 491 000
 {James.Nord,Kare.Synnes,Peter.Parnes}@cdt.luth.se

Abstract

This paper presents an architecture for location aware applications, where positioning sources such as GPS, WaveLAN and Bluetooth devices can be seamlessly interchanged or even combined to achieve a more accurate positioning service with a higher availability than a single positioning source could provide. The architecture also supports peer-to-peer communication to allow clients to interchange position information over a local wireless network such as Bluetooth or WaveLAN. This enables a user to use other users position sources if their clients are close enough. The position information can be used directly by an application or be combined with habitual and other contextual information to achieve more personalized applications.

A generic positioning protocol for interchange of position information between position sources and client applications is introduced and different techniques for merging of position information are presented. The interfaces for an application to access the platform and the platform to communicate with positioning sources are also discussed. The paper finally touches on privacy issues and outlines a schema for handling positioning information by using contracts that are easily maintained and controlled by the user.

1. Introduction

Mobile applications are a rapidly growing field of software technology, where mobile devices such as hand-held computers with wireless network connectivity enable a new set of context-aware software applications. The context of usage can be determined by analyzing an individuals patterns in combination with information about the physical locality of the user, which is also stated by Nelson [5]. This paper describes an architecture for location-aware applica-

tions where the context of usage enhances the use of mobile devices.

We assume that in the not so distant future people will use mobile devices to manage their day-to-day life and these devices will inherently be connected to the internet by wireless communication technologies such as GSM, GPRS, UMTS, WaveLAN, HiperLAN or Bluetooth. These technologies can be used to position the mobile device and thus enable position-based look up of services available in the network. That is, a service should not be available unless the user is within a certain context determined by time, location and usage patterns. Mobile applications differ in this respect from conventional applications, as the location of a mobile device is dynamic.

Therefore, there is a need for a mobile positioning platform that not only encapsulates different positioning techniques but also offers additional combined positioning techniques. This would mean that a mobile application does not need to handle all techniques available but instead could use a collective service, the mobile positioning platform. The platform should also provide support for accessing the common services required by the majority of mobile applications such as map retrieval and service look up.

This paper describes not only how different positioning techniques can be used depending on availability (by using the most accurate positioning technique available at a given time), but also describes how these positioning techniques can be amalgamated into a more accurate service (by using a collective set of positioning information) and how mobile devices can exchange positioning information over limited range wireless ad-hoc networks such as Bluetooth or a WaveLAN Ethernet segment. Furthermore, it is important to recognize and make use of the changing sets of resources in their environment; in this case how a seamless interchange of positioning information can be supported by the mobile positioning platform. This paper also outlines a generic positioning protocol (GPP) for positioning information exchange between devices and other networked services. The protocol is a central part of the positioning plat-

*This work was sponsored by the SITI Internet3 program, the Måkitalo Research Center and the Center for Distance-spanning Technology.

form, where an XML scheme is used to describe the positioning information. The structure described by the XML scheme is then used throughout the positioning platform to represent position information.

Section 2 provides an overview of the architecture, followed by an outline of the positioning platform in Section 2.1 and the Generic Positioning Protocol which is covered in Section 2.2. Section 2.3 focusses upon the issues of security and privacy and leads into a discussion of the Map Service and the Service Infobase in Sections 2.4 and 2.5 respectively. Finally, Section 3 outlines our current implementation whilst Sections 3.1 and 3.2 present two prototype applications using the platform.

2. Architecture

The architecture is divided into four sections: the Positioning Platform, the Privacy and Security handler, the Map Service and the Service Infobase. An overview of the architecture is depicted in Figure 1. The positioning platform provides abstraction from the positioning devices used; it is responsible for collecting the data from all of the positioning modules and combining it into a single position and accuracy (Section 2.1). The privacy/security handler (Section 2.3) provides an access point to query the application whether to grant or deny any requests that come to the platform to obtain the users location. The map service (Section 2.4) provides methods for obtaining map information whilst the service infobase (Section 2.5) abstracts the process of locating services within a specified context.

Applications													
Privacy / Security	Positioning Platform							Map Service	Service Infobase				
	GPS	DGPS	MPS(GSM)	MPS(GPRS)	UMTS	Bluetooth	IR	WaveLAN	HiperLAN	SB MapService	Gula Sidorna	XML database	WWW Search

Figure 1. The Alipes Architecture

2.1. Positioning Platform

The positioning platform collects the positioning data from the different positioning modules and merges the information to form one virtual device with additional qualities. The platform is capable of utilizing both push and

pull devices. A GPS device that reports information back with certain time intervals is an example of a push device, whereas an Ericsson Mobile Positioning System (MPS) [3] device is an example of a pull device that retrieves MPS information over the internet on request. For each physical positioning device used within the platform a simple software module needs to be implemented to communicate with the device and to translate the devices native position format into a common format detailed by the Generic Positioning Protocol (GPP) (Section 2.2) used by the platform.

As each device has different accuracy and position segment characteristics it is possible to find a new position area that is overlapped by each individual position and with a greater degree of confidence. In the example shown in Figure 2 a GPS device reports that it is within an area bounded by box A and an MPS device reports that it is within the area B which is bounded by two circles. With this knowledge, the positioning platform could calculate a more accurate position, for example the position bounded by the intersection of A and B.

It is not always possible to precisely compute the intersections from the bounds of the devices due to the different characteristics of the various positioning devices. In some cases such as a high end portable laptop, it could be possible to compute this intersection, whereas on a portable handheld such as an iPaq computing the intersection could be cumbersome and utilize too much computational effort. In such cases it could suffice to report just the position with the highest accuracy, to just merge the two most accurate positions or to use a less accurate merging technique. It is therefore important that the platform can be optimized for the underlying host characteristics and needs of the programs utilizing it.

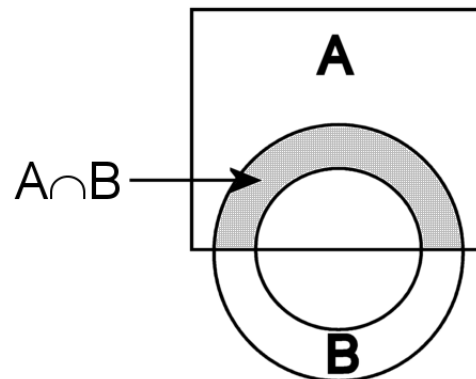


Figure 2. Calculating a more accurate position from two sources

As well as merging positioning information from lo-

cal devices, the position can also be augmented with information from other non-local position platforms via limited range ad-hoc networks such as a WaveLAN segment or Bluetooth networks. When any positioning information is sent over a network, the positions accuracy is changed to be the sum of the current accuracy and the maximum transmission range of the transmitting device. It is the duty of the device transmitting the position information to change this as in some transmission devices the range is dependent on the transmitting power which the receiver has no knowledge of.

In the example shown in Figure 3, two devices are sharing their position information over a Bluetooth network where both devices have a maximum transmission range of 10 meters. Device C has a position $P_{C(Global)}$ where $P_{C(Global)} = P_{C(Local)} \cap P_{C(Net)}$ and $P_{C(Local)}$ is obtained through a GPS device, whereas device D has a position $P_{D(Local)}$ received from a WaveLAN positioning server. Device C calculates the position $P_{C(Trans)} = P_{C(Local)} \pm (10m^*)$ to send over the network and transmits it. Device D can then calculate a new position for itself $P_{D(Global)} = P_{D(Local)} \cap P_{C(Trans)}$.

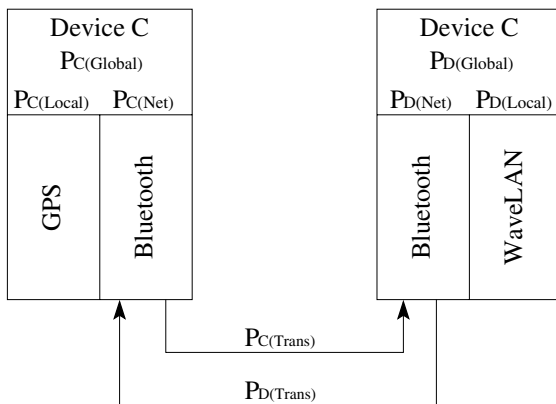


Figure 3. Peer-to-Peer position sharing

We have chosen to state that only locally trusted position information will be sent out over the network (Section 2.3), so any information learned via a peer-to-peer network will not be forwarded to other devices. The reasoning behind this is to reduce the effect that an erroneous positioning device or a rogue user could have on a system. If there is any conflict between the position that has been obtained over the network and the locally trusted position then the remote position information will be discarded.

It is also possible, where not directly supported by the underlying position device, for the platform to obtain a crude velocity vector indicating heading and speed. By taking the difference between the last known and current posi-

*The maximum transmission range of this Bluetooth link

tions a crude estimate can be obtained as to which direction the user is facing.

The positioning platform calculates the enhanced position and makes it available to the application as if it were itself a push or a pull device depending on the requirements of the applications.

2.2. Generic Positioning Protocol

The different positioning techniques have similar properties as they all report a position, but the protocols differ. A generic positioning protocol (GPP) should therefore support all positioning techniques and work as a common language for the positioning platform to communicate with external positioning sources. A source could be internal to the mobile device but external to the platform, such as an extra instance of the platform. This protocol should also be used against these sources. The structure of the protocol should also be kept within the positioning platform for communication with positioning modules internal to the mobile device (such as a GPS or Bluetooth device). Required features for the protocol are:

1. Structured and hierarchical format, for simple parser implementation.
2. Humanly readable, to aid debugging.
3. A few simple message types:
 - (a) Capability request
 - (b) Capability reply
 - (c) Data request
 - (d) Data reply

Due to the requirements that it be both humanly readable and in a structured-hierarchical format we have chosen to implement this protocol in XML. This also brings added benefits, as XML parsers are widely available for many different operating systems and architectures. The capability request and reply is used to determine the properties of the device, such as general accuracy, supported datum planes and coordinate systems and device type.

The main sets of messages are however the data request and reply messages. Only data reply messages are used for push devices, while both data request and reply messages are used for pull devices. The data request and reply messages have some common properties with the Ericsson Mobile Positioning Protocol [3], the Geography Markup Language (GML) [9] and the National Marine Electronics Associations (NMEA) 0183 standard. The GPP message structures are based on XML, but may be represented in other formats as well, for example, a request message could be represented as an HTTP query.

An example use of the protocol is between the platform and a WaveLAN positioning server, as in the RADAR system [1]. The client application queries the server to retrieve the position of the mobile device. The query includes information about the MAC address for the mobile devices together with additional information such as username and password.

2.3. Privacy and Security

It is vital to maintain the privacy of the users otherwise they might feel monitored or that the information would be used for unintended purposes. Want, et al. showed in their study of an active badge location system [15] that privacy issues are central and stress the importance of the ability to turn off positioning and that the user is informed of how the position information is used and/or logged. Thus a system like this therefore requires control of access to position information and that a user can see not only who has tried to locate her but also how often that has occurred. The questions are how to restrict access to positioning information in general and how the positioning system should be implemented in order to allow different levels of access.

Hence, the users application owns its position and that any external service has to request permission from the users application in order to obtain their position. It is therefore possible to query the platform for a position, but it will automatically result in a request to the application. The application is then free to accept or deny the request or to choose to ask the user if they wish to permit the query. Queries can be accepted once or for a longer period depending on a set of criteria defined by a contract between the user or application and the peer conducting the query. Currently implemented criteria include the ability for a service to query the platform a fixed number of times, unlimited queries in a specific amount of time, or any combination of the above two.

The user could also delegate the right to grant access to the users position to a secondary party, such as a WaveLAN positioning center, by constructing a contract. The contract will be limited by the users selection of criteria. All queries to the mobile device would then be redirected to the secondary party, which will follow the scope of the contract for all queries about the users position.

The user should also be able to deny all access to his position in a simple way, independent of any issued contracts to secondary parties. This could be done by revoking all contracts or to add additional criteria that override all others (i.e., deny criteria should take precedence over grant criteria).

Another issue is how to locate a users mobile terminal and thereby a users position. We have chosen a schema that is similar to the Session Initiation Protocol (SIP)[7], where

the IP address of a terminal is stored at a location server. In this case, the location server not only stores user information as defined by SIP but also position information restricted by contracts. This allows a service or application to first locate the source of position information for a user, either by querying the contracted location server or the mobile terminal if no contract exists. Note that using other directory services like the Lightweight Directory Access Protocol (LDAP) [16] instead of a schema similar to SIP would be as good.

There are however also other possible threats. The system could easily be compromised by false positioning sources, such as a Bluetooth beacon with a false position. Therefore, the sources should be separated into two groups, trusted and non-trusted positioning sources. The trusted sources could be a GPS module or an MPS position obtained through an MPC server [3] that the user can authenticate, while the non-trusted sources could be external sources such as other mobile devices within WaveLAN or Bluetooth range that have no method for authentication. In this case the positioning platform will prioritize trusted sources.

Another threat, not considered in detail here, is that the system could be compromised in other ways as well, for instance always possible to trace a general position of a mobile terminal since it has a network connection. This means that commonly found tools such as the Unix traceroute command will be able to achieve a general positioning of a mobile terminal. The user will also be traceable if he accesses local hardware such as an automated terminal for buying tickets or withdrawing money at a cash machine. The privacy protection covered in this paper needs to be complemented by more general protection schemes such as the IP Security Protocols [6].

2.4. Map Service

The generic map service is defined by a XML/HTTP interface that allows an application to retrieve maps for a certain position expressed in different geodetic datum planes and coordinate systems with additional parameters such as geodetic datum, coordinate system, map size, image type, scale and orientation. This scheme will help keep the service generic, as it is simple to change the underlying map database and retrieval system.

The current implementation uses a map database located at Luleå University of Technology (LTU), but could easily use other map databases with a different content and be located elsewhere such as the Telia Yellow Pages (Gula Sidorna) in Sweden. The map database at LTU is implemented to return the best available map depending on scale of the requested map. Specifically, queries with a scale set to smaller than 1:50000 return a map from one map series

(e.g. Röda kartan) while all other queries return a map from another map series (e.g. Ekonomiska kartan). The motivation behind this is that the user should not select the map series since he rarely knows the suitable scale-range for a certain map series.

2.5. Service Infobase

The service infobase provides methods for finding published services by searching a database or the internet for information that matches the criteria. The service descriptions can be represented as XML documents, where the criteria are represented as fields in the XML document. Similar ideas have been expressed in [8] which uses the Resource Description Framework (RDF) [10] as a basis for service description.

An example could be that a user searches for all the restaurants in the immediate vicinity that serve Italian food. The application would look in the service infobase for services that match and then present the location of each restaurant on a map. The user could then access additional information about the matches, such as if there is a 'how-to-get-there-guide' or even if the menu is available.

3. Implementation

The current client side implementation is written in Java and C/C++, for a StrongARM PocketPC (Compaq iPAQ 36xx) with Personal Java 1.1. The server side implementation is written in Java 1.2, Visual Basic and Perl. Currently available are software modules for GPS and WaveLAN positioning, as well as a map look up service (SB MapService). The GPS module includes support for parsing NMEA0183 [11] messages that are read from a serial port. The WaveLAN positioning server supports querying a set of WaveLAN base stations for connected MAC addresses and information in the location fields, which enables the server to return a rough position fix for a connected user. Work is ongoing to add software modules for MPS and Bluetooth positioning. The work on the Service Infobase has started, but far from completed.

Within the Alipes project we have created a mobile positioning platform and are developing prototypes to study the use context-aware applications. A set of prototypes using the positioning platform are being developed: FriendFinder, a tool for locating friends on a map, GeoNotes [12, 14], a tool for posting comments (similar to post-it notes), and BusLocator, a tool for locating the next suitable bus.

These prototypes will be evaluated within the eStreet project [4] by having approximately 20 users from the general public evaluate the prototypes. The goals are to understand how to develop context-aware applications based on positioning information and to understand what services

should be offered (especially in regards to privacy issues such as access control to a users position). The positioning platform will help us to rapidly develop, deploy and evaluate new applications.

3.1. FriendFinder

The FriendFinder application is based on a simple map navigator where the positions of registered friends are marked on the map (see Figure 4).

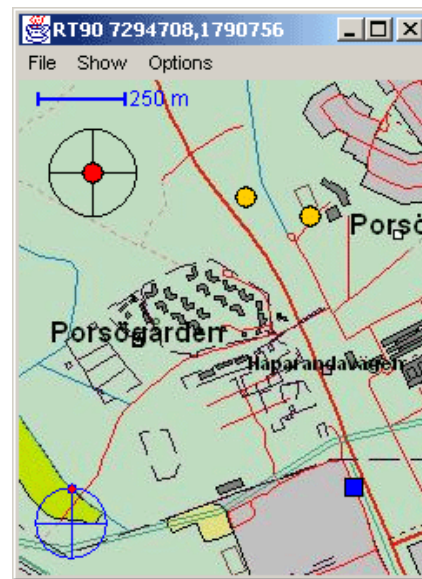


Figure 4. The FriendFinder application

The current position is marked by a red circle within a circled cross, friends positions are marked by orange circles and fixed positions are marked as blue squares.

3.2. GeoNotes

GeoNotes is a system developed at the Swedish Institute of Computer Science. The system allows users to annotate physical locations with virtual notes, much like placing Post-It notes on walls around you (see Figure 5). It is similar to applications described previously in [2, 13].

4. Summary & Conclusions

Although there are many different globally used positioning systems, there are few initiatives to incorporate these into one seamless system. Therefore, this paper describes a novel architecture for location aware applications, where positioning systems (including peer-to-peer position information exchange) are combined to form a positioning platform with additional benefits not offered by the single

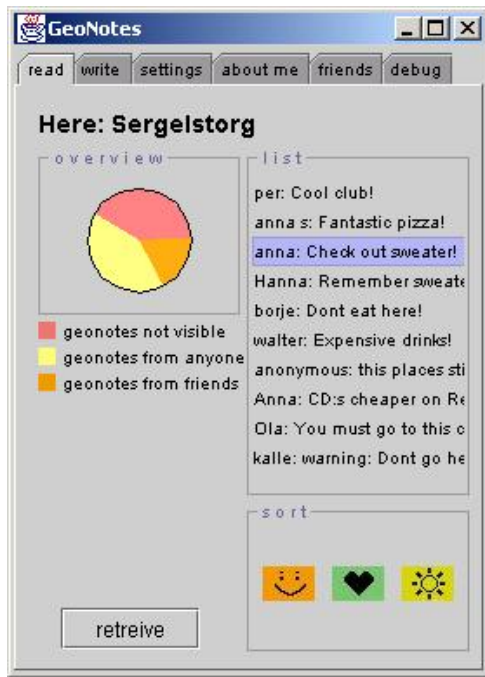


Figure 5. The GeoNotes application

positioning system. Applications benefit as they can be provided with more accurate positions and that they do not need to be rewritten for each new positioning technique.

We have also outlined a generic positioning protocol based on XML that enables exchange of position information as well as a map look up protocol. A service infobase is also introduced for location of available networked services. Finally, privacy and security issues have been discussed where the emphasis was on the users application to control access to the users position information.

5. Future Work

The architecture is currently in use by several research organizations and feedback from these organizations will be used to improve both the architecture and the implementation. Topics we have already identified as important are:

- an intelligent map cache with pre-loading support for times when you have limited network resources,
- a standard package/format for maps,
- definition and implementation of an intelligent system for search and filtering of networked services,
- a service description language,
- implementation and evaluation of a more advanced WaveLAN, Bluetooth and IR positioning technique,
- further study of privacy issues and
- support for anonymity towards services.

6. Acknowledgments

Thanks to Per Persson at SICS, Johan Andersson at Telia Research and Bo-Göran Stenman at Telia Mobile for valuable inspiration.

References

- [1] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF based User Location and Tracking System. In *IEEE Infocom 2000*, Tel-Aviv, Israel, March 2000.
- [2] P. J. Brown. The electronic Post-it note: A metaphor for mobile computing applications. In *IEEE Colloquium on Mobile Computing and Its Applications*, 1995.
- [3] Ericsson Mobile Positioning System (MPS), April 2001. <URL:<http://www.ericsson.com/wireless/products/mobsys/gsm/subpages/prod/overview.shtml>>.
- [4] The eStreet Project, April 2001. <URL:<http://www.estreet.lu/>>.
- [5] N. J. G. *Context-Aware and Location Systems*. PhD thesis, University of Cambridge, United Kingdom, January 1998.
- [6] I. I. S. P. W. Group. IETF IP Security Protocol, May 2001. <URL:<http://www.ietf.org/html.charters/ipsec-charter.html>>.
- [7] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg. SIP: Session Initiation Protocol, March 1999. IETF RFC-2543, <URL:<http://www.ietf.org/rfc/rfc2543.txt>>.
- [8] J. Hjelm and M. Nilsson. Position-Dependent Services Using Metadata Profile Matching. In *Inet2000 Conference*, Yokohama, Japan, 2000.
- [9] R. Lake, E. Keighan, J. Sharma, B. O'Rourke, and S. Johnson. Geography Markup Language (GML) 1.0: OGC Request 11, December 1999. <URL:<http://feature.opengis.org/rfc11/GMLRFCV1.0.html>>.
- [10] O. Lassila and R. Swick. Resource Description Framework (RDF) Model and Syntax Specification, February 1999. W3C Recommendation, <URL:<http://www.w3.org/TR/REC-rdf-syntax/>>.
- [11] N. M. E. A. (NMEA). NMEA 0183 interface standard, July 2000. <URL:<http://www.nmea.org/0183.htm>>.
- [12] H. Nyström and A. Sandin. Social Mobile Services in an Open Service Environment - an Overview, analysis and implementation. Master's thesis, Uppsala University, 2001.
- [13] J. Pascoe. The stick-e note architecture: Extending the interface beyond the user. In *International Conference on Intelligent User Interfaces*, Orlando, Florida, USA, January 1997.
- [14] P. Persson, F. Espinoza, and E. Cacciatore. GeoNotes: Social Enhancement of Geographical Space. In *ACM's Conference CHI01*, Design Expo Category, Seattle, USA, April 2001.
- [15] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location System. *ACM Transactions on Information Systems*, 10(1), 1992.
- [16] W. Yeong, T. Howes, and S. Kille. Lightweight Directory Access Protocol, March 1995. IETF RFC-1777, <URL:<http://www.ietf.org/rfc/rfc1777.txt>>.