



White Paper

User Location in Heterogeneous Mobile Networks

Authors: Maarten Wegdam, Jeroen van Bommel, Ko Lagerberg
Bell Labs Advanced Technologies EMEA – The Netherlands

Project: AlbatrOSS – an European Union IST FP5 project - <http://www.ist-albatross.org/>

Contact: {wegdam,jbommel,lagerberg}@lucent.com

Version: January 2004

ABSTRACT

Fourth generation mobile networks will allow end-users to roam over different network technologies, such as UMTS, CDMA2000 and Wi-Fi. These mobile networks have the ability to determine the location of the end-user, which is information that can be used by mobile applications to adapt their behavior. Each of the network technologies has its own way to determine the user location, and has its own way to provide this information to the end-user and/or mobile application. Developers of mobile applications however should not be exposed to the peculiarities of the different network technologies, and should be offered a network technology independent way to obtain the user location. In this paper we propose a solution to provide developers of mobile applications with user location information over heterogeneous networks that are managed by different parties. Since user location is privacy sensitive information, we explicitly address end-user control over the access to user location information. We focus on UMTS and Wi-Fi networks, and have prototyped our solution on these networks.

KEYWORDS

Wi-Fi, WLAN, UMTS, 3G, AlbatrOSS, User Location, Parlay, OSA, Parlay X, Location Based Services, positioning method, roaming



1 Introduction

Current second and third generation mobile networks allow roaming between networks of different network operators. With fourth generation networks (4G) the roaming concept is extended by also allowing roaming between different networks technologies. This results in a federation of partly overlapping mobile networks that are multi-party and multi-technology. The user can access mobile applications as long as there is coverage by at least one mobile network, and in case there is coverage by several mobile or wireless networks the user can choose between them based on costs, bandwidth, etc.

For both end-users and the developers of mobile applications it is essential that they are not exposed to the complexity of the multi-party and multi-technology network. End-users want to use mobile applications without being concerned with technical details of the underlying network technologies. Developers of mobile applications want to develop applications that work for a large variety of network technologies, both current and future, and in general do not have expertise on specific network technologies or how the networks are deployed by the different network operators.

A large class of mobile applications uses the ability of mobile networks to determine a mobile terminal's location to adapt their behavior to the location of the user. This is sometimes referred to as Location Based Services. Such mobile applications require an interface to obtain the location information. There is standardization to enable this for certain network technologies, specifically 3G networks [OSA], and there are products to support this [ISG]. However, what we need is an integrated way to determine the user location, that integrates point solutions that only work for certain network technologies, or for certain network operators.

This paper describes a way to provide such an interface to mobile applications, which they can access to determine the user location in a 4G scenario. Our solution uses the point solutions that the different network operators and the different mobile and wireless network technologies offer, and hides the complexity of these point solutions for the mobile application. Besides describing the issues that have to be addressed to determine the location of a user in a 4G scenario, we describe alternative solutions, and zoom in on one of the specific solutions we selected. We also describe a prototype of this solution. We have limited the scope of our research and our prototype by focusing on two specific network technologies: Wi-Fi and UMTS. We however believe that our research can easily be applied to other network technologies as well.

The research that we describe in this paper has been done as part of AlbatROSS project (www.ist-albatross.org), which is sponsored by the European Union (IST FP5).

1.1 Structure

Section 2 gives an overview of the context for this white paper, i.e., roaming in heterogeneous networks, and the different stakeholders that are involved in this. Section 3 explains how location can be determined in the network technologies that we focus on: UMTS and Wi-Fi. Section 4 discusses the requirements for providing a solution for determining location for heterogeneous networks. In Section 5 we propose two alternative architectures, compare them and select one to prototype. Section 6 contains a more detailed description of this prototype, and discusses several design decisions that we made. We end this paper in Section 7 with conclusions and future work.

2 Roles in Location-based Services

For the purpose of this paper we assume a number of stakeholders that play a role in the offering of location-based services. These roles are depicted in Figure 2-1.

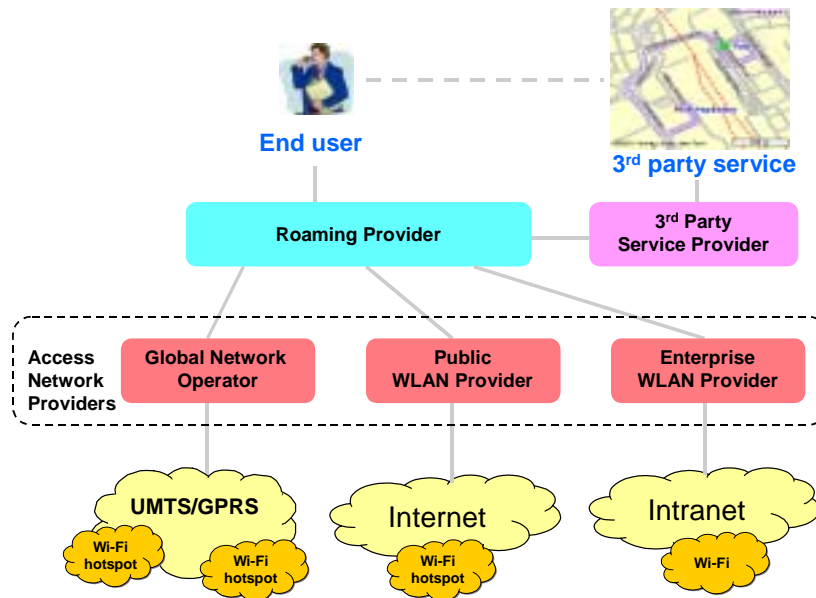


Figure 2-1 Stakeholder roles for location-based services

End-user

The end-user's role is twofold. He/she is both the subject matter for a location aware service and a possible consumer of the (application that uses) location information.

Access network provider

Generally, the access network provider is a source for user location information. Depending on the used network technology, the reported location is based on the point of attachment of the end-user to the access network. This location information is offered to the stakeholders higher up in the network hierarchy.

We distinguish the following three access network provider types:

- *Global network operator*, operating a mobile network with user location capabilities. Typically this is a 2.5 or 3G network, possibly extended with wireless LAN hotspots.
- *Public Wi-Fi provider*, operating one or more public Wi-Fi hotspots. The hotspot provider may or may not have an affiliation with a global network operator, but there is no tight integration with the PLMN. In our model we assume the public hotspot provider is capable of providing user location information for end-users accessing a hotspot.
- *Enterprise Wi-Fi provider*, operating an enterprise network with one or more Wi-Fi networks each consisting of one or more access points. This provider's role is comparable to the role of the public Wi-Fi provider. Typical differences are security requirements and solutions (e.g. VPN), available services, and service charging, but for the sake of user location information there is not much difference. The enterprise is capable of providing this information.

Roaming provider

The roaming provider has multiple roles. First of all, it maintains roaming agreements with multiple access network providers, and aggregates these networks into a single – possibly heterogeneous – access network for end-users.

Secondly, the roaming provider may have agreements with other roaming providers, thereby extending the coverage area for its users. The agreement may include the exchange of location information of visiting users.

Finally, the roaming provider maintains agreements with end-users for the purpose of network and service access. Payment is typically monthly, and may be based on usage fees. The

roaming provider supplies the end-user with the necessary credentials to access the network and optionally value-added services.

The roaming provider offers to the end-user the possibility to personalize the services, e.g. create personal service profiles, control privacy settings, etc.

3rd party service provider (3PSP)

The 3PSP offers end-user applications, which can be location-aware. Examples are applications that provide the local weather, restaurant finder and a movie ticket reservation application.

All of these roles can be assumed by different parties, or they can be combined by a single party. For example, an arbitrary telecom operator could unify all roles (except the end-user role). Alternatively there can be parties that combine one or more roles. In a typical case, a 3G mobile network operator would provide a 3G access network with user location support and embedded Wi-Fi hotspots. Additionally the operator can have an affiliation with an independent public Wi-Fi hotspot provider, allowing its users to roam not only between network technologies, but also between administrative domains. For 3rd party service providers, the operator can offer a single point of access to its user location service by deploying the user location mediator. This way Location Based Services can be offered to all roaming users, without having to worry about the network domain a user is in.

3 Location in UMTS and Wi-Fi

Although mobile phones and PDAs can be extended with a Global Positioning System (GPS) receiver, which can determine the location with an accuracy of tens of meters, this is unlikely to happen in the coming years because of cost and battery life considerations. In addition, GPS can have a large startup-time (at least 30 seconds to a few minutes) and does not work inside a building or in densely populated urban areas.

Currently, the only viable alternative is to rely on the network to determine the location of end-users. In this section we will discuss this approach for the two network technologies we are focusing on in this paper: UMTS and 802.11b (Wi-Fi) networks.

3.1 UMTS

We first give a brief overview of the different methods for determining location, and then discuss the Open Services Access (OSA) specification, which is standardized by Third Partnership Project (3GPP) and allows mobile applications to access this location information.

3.1.1 Location Positioning Methods

There are three methods standardized for UMTS to determine the location of a mobile phone [UMTSPos, Zhao02].

The cell ID based positioning method is the most trivial indication of the user location, and does not require specific functionality in the UMTS network (or GSM/GPRS for that matter). It is however not very accurate. The average size of radio cells in UMTS can vary from 800 meters radius in dense urban areas to about 6 kilometers in rural areas [Samuel03]. With additional measurements it is sometimes but not always possible to achieve a higher accuracy than the cell area [Zhao02].

The observed time difference of arrival (OTDOA) positioning method uses observed time differences between mobile phone and close-by base stations (of which the exact location is known) to determine the location. The measurements of different base stations are used to triangulate the location. The accuracy of the OTDOA positioning method varies depending on the actual location of the mobile phone within the cell; especially if a mobile phone is close to one of the base stations, it may be difficult to *hear* the two other base stations needed for the triangulation. Although some controller products implement this method, it is not yet widely deployed.

A third method is the network assisted GPS method (AGPS), which requires the mobile phone to be fitted with a GPS receiver. Although this method has the above-mentioned cost issue, it does offer some benefits over a GPS-only scenario, such as when the GPS does not have clear sky visibility, faster startup time and lower power consumption (since the GPS can be offline more often).

In summary, the UMTS network is capable of determining the location of a mobile phone, but the accuracy depends on the used measurement method and the actual location of the phone within the cell, and will vary from hundreds of meters to only a few meters. In addition to the location of a mobile phone, the UMTS network can also estimate the accuracy of the reported location. UMTS networks can also deploy proprietary positioning methods (especially if they do not require changes to the mobile terminal) instead of or next to the above three standardized positioning methods,

For GSM and for cdmaOne and cdma2000 three similar positioning methods have been specified. For GSM networks the OTDOA-like positioning method is called enhanced observed time difference (EOTD). For cdmaOne and cdma2000 the OTDOA-like positioning method is called advanced forward link trilateration (AFTL) [Zhao02].

3.1.2 Open Services Access, Parlay and Parlay X

The Open Service Access (OSA) specification [OSA] is a 3GPP standard that allows third party mobile applications to obtain the location of a certain user. This is part of the OSA User Location Service. There are also other OSA services and capabilities, but this is outside the scope of this paper.

OSA is defined by 3GPP. OSA was designed to be “technology agnostic”, in the sense that it should be equally suitable for 2G, 2.5G and wire line networks, as it is to 3G networks. In reality however, OSA is biased towards 3G and to a lesser extent 2G and 2.5G mobile networks. Parlay [Parlay] is a closely related specification, but is instead specified by the Parlay industry consortium, and focuses more on fixed networks. The OSA and Parlay standardization is mostly done collaboratively, and the standards as a consequence are mostly identical.

OSA is specified in a middleware-technology independent manner, using OMG IDL [CORBA]. CORBA is however the most used middleware technology for this. To address the market of web services, the OSA/Parlay API was simplified and adapted to use SOAP [SOAP] as transport protocol [Lagerberg02]. This simplified SOAP version of the specification is called Parlay X.

3.2 Wi-Fi

Wi-Fi (standardized as IEEE802.11b) is a wireless LAN technology that is gaining momentum. So-called hotspot providers deploy Wi-Fi access points at hotspots, such as train stations, coffee shops and airports. There is no nation wide Wi-Fi coverage, this is not feasible due to the small area one access point can cover. Also, every hotspot provider is expected to deploy a limited amount of hotspots, and for coverage at a wider range of places an end-user will need to use access points of different hotspot providers.

Determining the location of an end-user using the Wi-Fi network can be done using similar triangulation positioning methods as in the case for UMTS. This is however not very common. Since a Wi-Fi access point (AP) has a range of only a few hundred meters in the most optimal case, more often only tens of meters [Prasad00], knowing which access point an end-user is connected to already pinpoints the location relatively accurately (compared to the UMTS or GSM case). Of course, the location of this access point has to be known. One way of implementing this is to maintain a simple database, typically per hotspot provider, containing the location of every access point. To communicate this location to the client device, one option would be to extend the RADIUS protocol – used for authentication – with a ‘location’ attribute (e.g. as part of the Extensible Authentication Protocol (EAP) Message) to be forwarded transparently by the AP. Alternatively the location can be stored in the access point itself, and sent to the mobile terminal, for example by extending the DHCP server that is already commonly embedded in access points. This approach would not solve the accessibility of location information by third party applications. In any case there is no widely accepted or standardized interface for this.

To obtain a more accurate position than static per-AP provisioning methods, it is possible to use triangulation methods much like those proposed for the UMTS network (see above). A common method is based on signal strengths to multiple (≥ 3) APs, which can be obtained from both the AP and the client device. For an AP, one would use SNMP (Simple Network Management Protocol) to query the AP for the currently perceived signal strength for a particular client. Although there is no standard SNMP MIB for APs that implements this, many

vendors provide extensions that make this possible (e.g., Orinoco and Cisco). On the client side, the operating system commonly provides this information through an API (e.g., Windows WMI and NDIS 5.0, Linux per device driver).

4 Requirements

Our goal is to be able to determine the current user location in heterogeneous mobile networks, and provide this user location to applications running at a 3rd party service provider and/or on the mobile terminal. The location should be coded in latitude/longitude or similar format, contrary to access provider proprietary location identification such as Cell Id. In addition to the location itself, the accuracy of the location and the time at which this location was determined should also be provided.

4.1 Classification of Location-based Applications

Before discussing the requirements for a solution that provides location information in a 4G environment, we first present a high level classification of location-based applications.

There are two ways to communicate location coordinates: *push* and *pull*. In *push* type applications location updates are sent to a receiver either periodically or triggered, for instance when the difference with the last known position exceeds a certain threshold distance or a specific area is entered / left. *Pull* type applications send a request for location to a server or directly to the device being located, and receive a reply.

We distinguish roughly four types of distribution architectures. Standalone *client-side applications* reside on the user's device. Mixed *client/server-side applications* have some application logic on the user's device but require a server in the network for proper operation. *Server-side applications* have only thin presentation logic on the user's device. Finally, *peer-to-peer applications* operate between a decentralized set of cooperating client devices.

The target to be located can be the *user* himself, or *other* users or objects.

In the table below we only consider applications that require in some way the use of a radio network. Other classes of location-based applications, such as shipment tracking and car navigation with in-vehicle storage of maps, are left out.

Table 4-1 Types of applications

Example application	Push / Pull	C, C/S, S, C ⁿ	User / Others
City navigation with Points of Interest	Pull	Client/Server	User
Friends finder	Either	Client/Server or Peer-to-peer	Others
Location specific web sites	Pull	Server	User
Workforce tracking	Push	Client/Server	Others

The characteristics of applications impact the choice of architecture. For example, a workforce tracking application would benefit from an architecture in which the location can be (logically) centrally requested.

4.2 Requirements

The following requirements need to be addressed to be able to provide location information for a 4G network:

- *Network technology transparency* - the type of network technology the user is currently using should be hidden from the developer of mobile applications.
- *Operator transparency* - the access network provider the user is currently using should be irrelevant for the developer of the mobile applications.

- *Privacy* - a solution should provide the means to let the user control which mobile application gets access to his location. This should not be a specific solution per access network operator (see transparency requirement), but an integrated solution.
- *GPS* – use GPS if mobile terminal has this feature.
- *Air communication* – minimize communication over the air, since this is slow and expensive, use fixed line communication whenever possible.
- *Always available* – mobile terminals are often out-of-range or switched off, a solution should allow access to the last known location even if this is the case.
- *Minimize impact* – minimize the impact on existing network elements, protocols and interfaces
- *Suit different application types* – the solution should work for all the different application types described in Section 4.1.

5 Alternative Architectures

Depending on the application context and other factors there are several choices to be made for any architecture that supports location-based services. Figure 5-1 illustrates some of the architectural choices. We show that there are three major choices to be made:

1. What are the reference points for positioning: satellites or base stations? A hybrid solution is also possible here, but is omitted for simplicity.
2. Where is the user location determined: in the terminal or by a network element?
3. In case the user location is determined in the terminal, is this user location then sent to the network, or kept locally in the terminal?

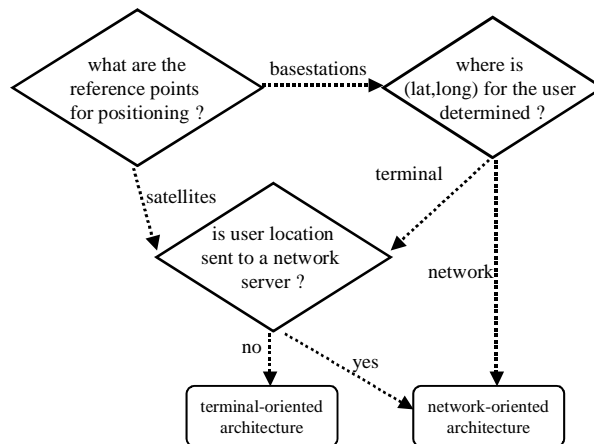


Figure 5-1 Architectural choices

The above three architectural choices result in two different alternative architectures: the terminal-oriented architecture and the network-oriented architecture. For the network technology and network operator transparency to work, the knowledge to which network(s) a user is currently connected has to be embedded in generic components (as opposed to exposing and embedding this in the application components). This division in terminal and network oriented architectures indicates where this functionality resides.

We will describe these two alternative architectures in the next section, and evaluate and compare them in Section 5.3.

5.1 Terminal-oriented Architecture

In the terminal-oriented architecture it is logic executed on the terminal that implements the transparency towards the client applications (i.e. which network technology and which network access operator is currently used, and where location information comes from). This is

depicted in Figure 5-2. Since the user location issue is closely related to identifying the user, we also show the user authentication in this figure.

Client-side application logic can access the location through a local API on the terminal. It can then use some proprietary protocol to send this location information to the server-side application logic.

The different network-technology-dependent mechanisms to get the user location are also embedded in the terminal. E.g., in case a DHCP attribute is used to pass the location in a Wi-Fi network, there is software installed on the terminal to read this location field. A condition for this to work is that there is a limited set of de-facto or de-jure standard mechanisms to get the location from the network, which are currently not or only partly available. An alternative approach would be to dynamically upload the software to the terminal to get the user location from the network, which could then use proprietary mechanisms. However, this would require significant standardization on the execution environment of this software in the terminal. For terminals supporting J2ME, the Java Community Process released the Location API for J2ME to do this [JSR179]. The level of industry support for this API is unclear.

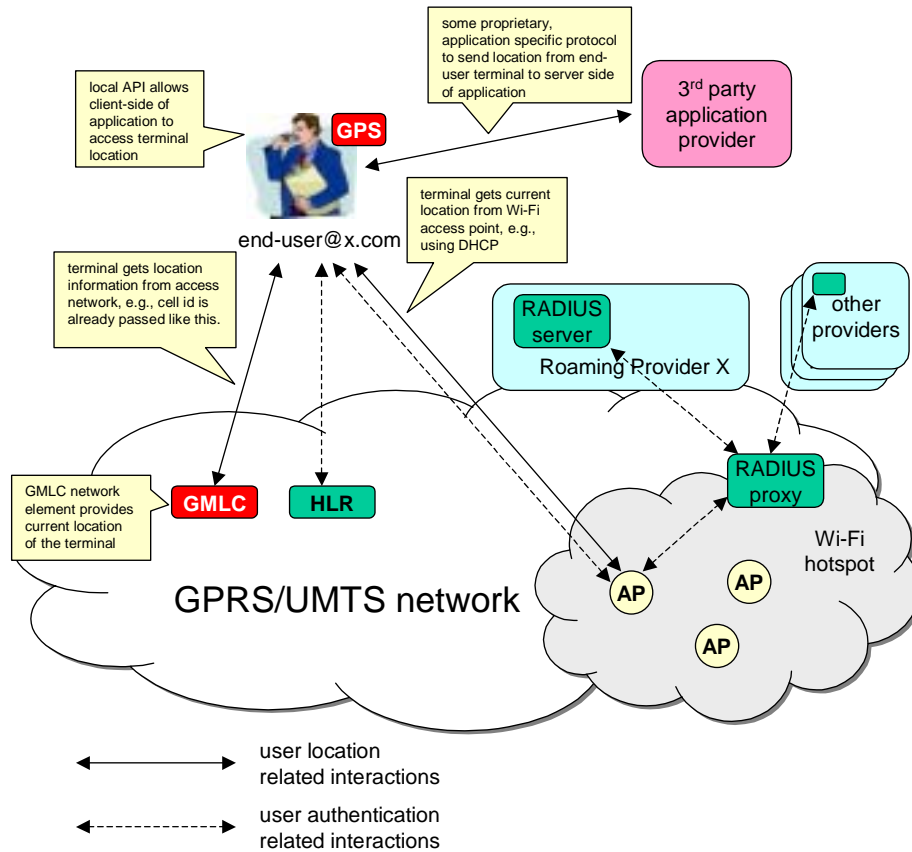


Figure 5-2 Terminal-oriented architecture

Privacy can be ensured by allowing end-user control over which applications get access to the user location, and for what purpose. This is relatively easy to implement in a terminal-oriented architecture since this can be done locally in the terminal. A simple implementation would present a pop-up to the user whenever an application requests the user location, asking if this should be allowed. More sophisticated implementations would allow some form of automation, e.g., policy-based logic that automatically grants permission to trusted applications.

5.2 Network-oriented Architecture

For a network oriented architecture we need to know in which network a user is currently located. Instead of having this logic in the terminal, it resides in the network or – to be more specific – with the roaming provider.

In this architecture, we assume that a user has a contract with a roaming provider who enables seamless roaming between GPRS/UMTS and Wi-Fi hotspots. To realize this, the roaming provider offers a service platform that includes a RADIUS server for authentication, authorization and accounting (AAA) of roaming users. When a user enters a Wi-Fi hotspot area, the authorization for accessing the Wi-Fi network is not directly done by the hotspot provider, but indirectly by the roaming provider. The hotspot provider forwards the user credentials towards the AAA server of the roaming provider, and awaits the authorization response. Figure 5-3 depicts this process using the dotted lines.

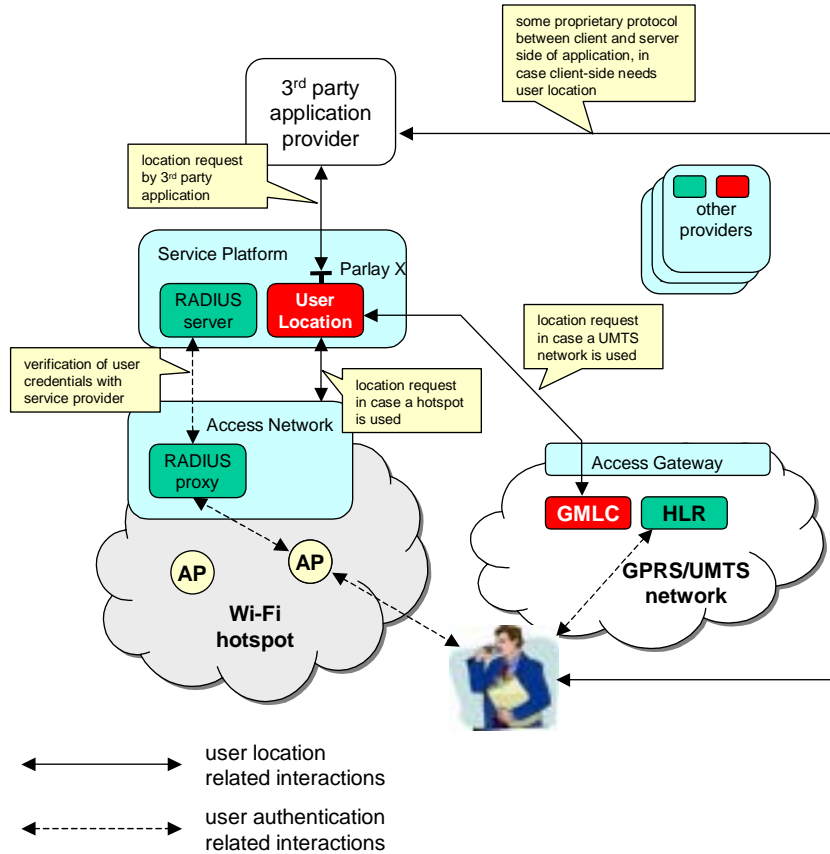


Figure 5-3 Network-oriented architecture

Since the AAA server located within the service platform is notified – due to the authentication – of the user’s current hotspot connection, it can direct any user location request to the appropriate hotspot provider. Third party applications can then contact the service platform for user location requests, which functions as an intermediary to either the current hotspot the user is connected to, or the GPRS/UMTS network. This is depicted with the solid lines in Figure 5-3.

5.3 Comparison

Below we compare the terminal-oriented and the network-oriented architecture, based on the requirements we identified in Section 4.2.

Table 5-1 Comparison between terminal of network-oriented architectures

Requirement	Terminal-oriented	Network-oriented
network technology and operator transparency	implemented in terminal, which makes terminal more complex and can reduce battery lifetime	implemented by the roaming provider
privacy	controlled by end-user via its	enforced by roaming provider on behalf

	terminal	of user, which requires the user to trust the roaming provider (and a possibility for the user to set his preferences)
GPS integration	can be done in terminal	more difficult, would require additional hardware
air communication	air communication for every time (server side of) application needs location	little to no air communication
always	no user location if terminal is out of reach or off	last-known location can be maintained, this can however be outdated and/or less accurate
minimize impact (assuming every network technology can determine location)	requires <ul style="list-style-type: none"> ▪ a standardized OS (or VM) API in mobile terminal for client-side application logic ▪ interfaces/mechanisms between terminal and every network technology to get the user location 	requires <ul style="list-style-type: none"> ▪ new fields in e.g. authentication protocols between access network provider and roaming provider to pass reference to User Location service in access network (or alternatively a provisioning interface offered by the roaming provider)
applications suitability	can cause unacceptable cost for push applications due to air communication, especially when periodic	less suitable for pure client-side applications

In summary, in a terminal-oriented architecture it is easier to implement end-user control over who gets the user location (privacy), to integrate with GPS and to implement pure client-side applications. The network-oriented architecture however requires less air network traffic, can better cope with terminals that are switched off, is faster, is better suited for push-type applications and requires less new standardization.

Based on this comparison we chose to detail and prototype the network-based architecture.

6 Proof-of-Concept

6.1 Prototype implementation

The network-oriented architecture has been validated by means of a prototype. Figure 6-1 depicts our solution in more detail. We use the Parlay X [ParlayX] standard as the interface between application and service platform, and as the interface between the service platform and the access networks.

The figure illustrates the dynamic behavior for two aspects of the concept: keeping track of the location of a roaming user (1, 2, 3,...), and a 3rd party application requesting the user's current location (a, b, c,...).

The following steps are performed:

1. As a prerequisite, the service platform registers its location update interface in a UDDI (Universal Description, Discovery, and Integration) registry, allowing access networks to discover it.
2. The user enters the Wi-Fi hotspot area and starts the Wi-Fi access procedure.
3. The access point requests the user's authentication credentials, and forwards these in a RADIUS access request to its local RADIUS server/proxy.
4. The RADIUS proxy forwards the access request to the RADIUS server of the user's roaming provider. When the authentication is successful and the user is authorized to make use of the hotspot, access is granted.
5. Upon successful completion of the access procedure, the RADIUS proxy of the Wi-Fi hotspot informs the hotspot's user location component of the user's presence in the network.
6. The UL component looks up the location update interface of the user's roaming provider in a UDDI registry (see also step 1). The result can be cached for increased performance.

7. The UL component informs the roaming provider of the network the user is in. This information includes
- a local user-id, to identify the user in the hotspot network, and
 - the URL of the Parlay X interface of the hotspot, where location information for this user can be collected.

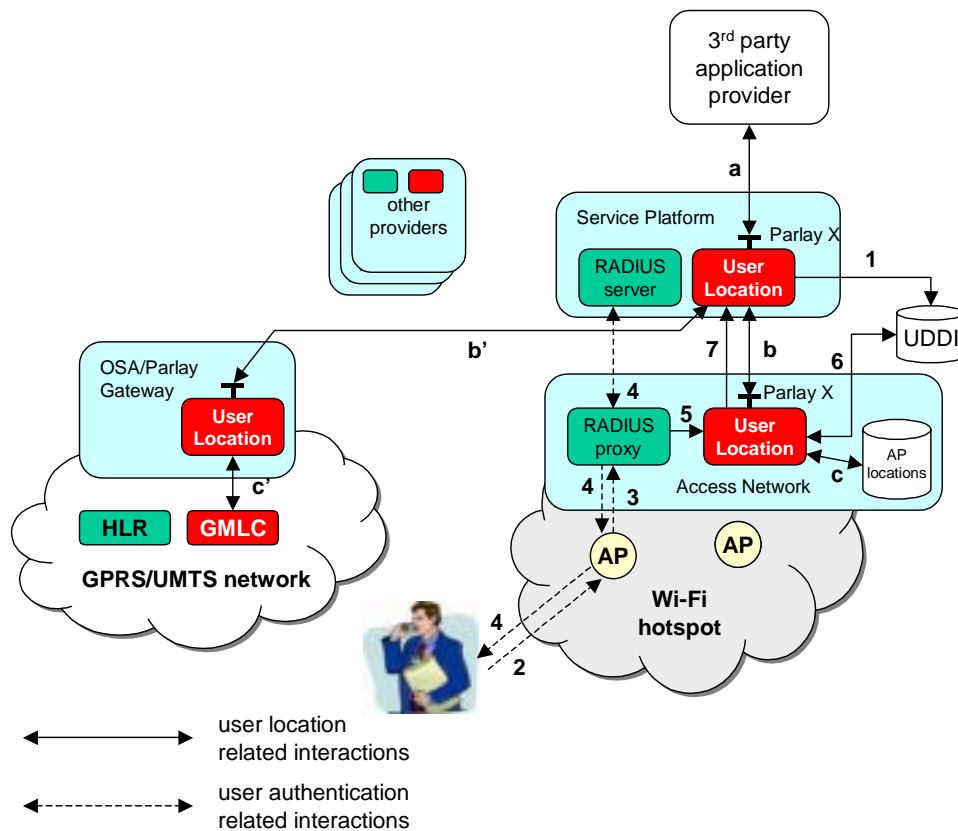


Figure 6-1 Network-oriented solution

When a 3rd party application requests the location of a user who is currently accessing a Wi-Fi hotspot, the following steps are performed:

- The application requests the location of a user from the Parlay X interface of the roaming provider.
- The roaming provider verifies that the end-user allows this application to know his location.
- The user location component of the roaming provider determines the current access network of the user, looks up the URL of the network's location interface and the corresponding local user-id for this interface, and issues a location request on this interface.
- The UL component of the access network determines the actual location of the user, by looking up the location of the associated access point in a local database. The location information is returned to the roaming provider, which forwards the information to the 3rd party application.

Alternatively, when the user is currently registered in a UMTS (or GPRS for that matter) network, the steps would be:

- The user location component of the roaming provider requests the location information from the access network. This can be through an OSA/Parlay gateway, Parlay X gateway, or any other proprietary or standardized location gateway.
- The user's location is determined by the network (e.g. GMLC) and returned to the roaming provider.

The prototype solution supports two interface flavors for obtaining user location information from an access network: OSA/Parlay and Parlay X. In a live network, other interface types

could be supported, depending on the capabilities of the network. The 3rd party however only needs to support Parlay X.

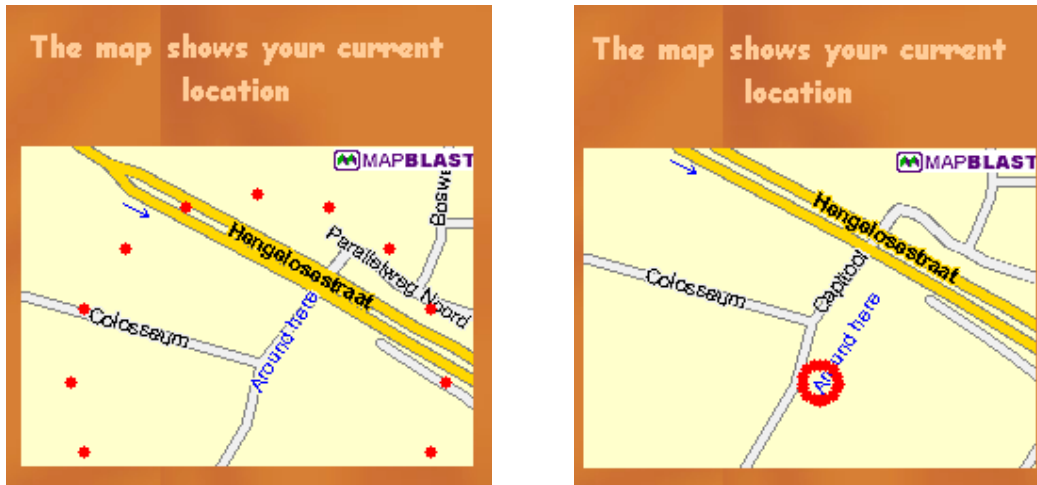


Figure 6-2 Accuracy of user location in UMTS (left) and Wi-Fi (right) networks

Figure 6-2 shows a snapshot of a sample location-based application that is part of the prototype. It illustrates the difference between obtaining the user location from a UMTS (left) and a Wi-Fi (right) network. The circle of red dots indicates the uncertainty of the position as determined by the network. In this case the position obtained from the Wi-Fi hotspot is more accurate. Other than this, there are no differences to the user, the application, or the service provider.

An issue of concern is the privacy of the end-user. The solution should ensure that only the parties that the user trusts (i.e. roaming provider and 3rd party service) are allowed to get the user's location. While OSA/Parlay provides a solid access control mechanism through the OSA Framework API, there is no such mechanism in Parlay X. The Parlay X standard prescribes that an implementation needs to define its own access control mechanism if required. Without properly addressing this issue our solution would be vulnerable to applications that spoof a valid identification token.

To partly address this issue, the prototype uses anonymous identifiers for the Wi-Fi Parlay X interface. When a user enters a Wi-Fi network, the user location component generates an anonymous id for this user, and includes it in the location update that is sent to the service platform (step 7 in Figure 6-1). When subsequently the service platform requests the user location, the anonymous id is passed as a parameter instead of an identifier that would reveal the user's identity. In this way, a malicious application that would somehow discover the location of this Web Service interface could make requests, but would get a response that it cannot link to any particular user.

The prototype also includes an access control mechanism on the Parlay X interface between 3rd party service and service platform, but this solution currently lacks adequate security as it is based on simple identifiers.

6.2 Other considerations

If a user is active in two or more networks simultaneously, the service platform could send a request to only one of them, or it can query more (even all) of them for the user location. Especially if we assume Wi-Fi and UMTS, the UMTS (or GPRS/GSM) coverage will typically be a superset of the Wi-Fi coverage, and thus the user's location could be obtained from either network. This does not mean that the user location information reported by each network is identical, in fact the Wi-Fi network might report a much more accurate location, or the UMTS network might not support user location.

In the prototype the (simulated) UMTS core network is treated as a black box, so in particular the roaming provider is not notified when a user is actually connected to the UMTS network (contrary to Wi-Fi hotspots). Therefore, the roaming provider must use polling to try and obtain

the user location. This inefficiency could be improved when an asynchronous notification service would be implemented in the UMTS core, such as OSA/Parlay UserLocation and/or UserStatus.

The prototype is part of a test environment that includes Mobile IP [MIP]. Although the solution described in this paper does not depend on Mobile IP, we could implement a somewhat similar but alternative solution: instead of using Wi-Fi authentication triggers, it is possible to use Mobile IP registration triggers to send updates to the roaming provider. An issue with this solution is that hotspot identification information is not available in standard Mobile IP registration messages, so they would either need to be extended or this information could be inferred (e.g. by noting the router from which the request came).

7 Conclusions

4G networks consist of a collection of heterogeneous mobile, or both mobile and fixed/wire line networks, and will allow seamless roaming between the different network technologies. Typical technologies are Wi-Fi and UMTS/CDMA2000, while multiple access network providers and roaming providers are involved. Location-based services are an important class of services that will be provided on these 4G networks, and these services need access to the user location in a transparent manner, i.e., independent of the specific network technology and network access operator the user is currently using. Important issues that have to be addressed when providing the user location include minimizing air communication (because this is slow and expensive), ensuring privacy and minimize the need for new standards.

Focusing on the representative network technologies GPRS/UMTS and Wi-Fi, we have sketched two alternative architectures to provide the user location in a 4G scenario based on where the functionality is located: terminal-oriented and network-oriented.

The main benefit of a terminal-oriented architecture is that ensuring privacy is easier to implement, and integration with GPS is straightforward. However, this requires more air communication, more processing on the terminal and extra standardization. The network-oriented architecture does not have these disadvantages, but it does require the user to trust the roaming provider to ensure its privacy. Which architecture is most suited depends also on the type of applications the user wants to use, and the terminal he is using. If the user has a 'fat' terminal with sufficient battery and processing power (e.g., a laptop), and he uses applications that run purely on the mobile terminal, the terminal-oriented architecture is better suited. For thin terminals for which part or most of the application logic is located with a 3rd party application provider, the network-oriented architecture will often be more suited.

We prototyped the network-oriented architecture for Wi-Fi and GPRS/UMTS networks. Our solution exploits the central role of the roaming provider, which is needed to allow roaming over heterogeneous networks anyway. The roaming provider offers the Web Services (SOAP) based Parlay X interface towards 3rd parties to obtain the user's location, while providing transparency for the various access network providers.

GPRS/UMTS networks have standardized mechanisms to determine the user location, for Wi-Fi networks this is not the case. We therefore have built our own user location functionality on top of the authentication mechanisms that are used between terminal, Wi-Fi access network and roaming provider. The authentication mechanisms are used by the roaming provider to know if – and if so which – Wi-Fi hotspot provider a user is currently using, so that the request for a user location can be directed to the appropriate hotspot provider. We exploit non-specified fields in the authentication protocols to avoid any need to change their specification.

7.1 Future Work

Below we mention some open issues that we have not addressed yet in this paper and/or our prototype.

We mentioned the privacy issue as an essential one for allowing 3rd party access to the user location. There are several issues related to this that need to be addressed, including the granularity with which the end-user should get control over the user location (per 3rd party application provider, per application, per specific request for user location), how can end-user control be implemented in such a manner that it is understandable for the end-user and how to implement push-type location information from the roaming or access network provider to the user to query the user for privacy-related permissions.

We did not address the issue of charging and accounting for the user location functionality. Since there are costs involved in developing, deploying and maintaining this functionality, someone has to pay for this. Possibly separate accounting and billing for this proves too cumbersome and expensive, and the user would only pay for this of part of his roaming contract with the roaming provider, but other revenue models are also possible and should be explored.

We have sketched some of the more important design issues when implementing user location determination for Wi-Fi networks, these issues could be explored in more detail and standardization could be pursued for this to facilitate interoperability (e.g., standardize the non-specified fields in the authentication protocols we use).

With respect to our prototype, we could add other network technologies, notably fixed networks. In addition, we could implement the terminal-oriented architecture, and possibly also integrate this with the network-oriented architecture.

References

- [EAP] PPP Extensible Authentication Protocol, <http://www.faqs.org/rfcs/rfc2284.html>
- [ISG] Lucent Technologies, MiLife™ Intelligent Services Gateway, http://www.lucent.com/solutions/mobile_apps.html
- [JSR179] Java Community Process, JSR-000179 Location API for J2ME, <http://jcp.org/en/jsr/detail?id=179>, Sept 2003.
- [Lagerberg02] Ko Lagerberg, Dirk-Jaap Plas, Maarten Wegdam, Web Services in 3G Service Platforms, Bell Labs Technical Journal, Published by Wiley Periodicals Inc., volume 7, number 2, (p 167-183), December 2002.
- [MIP] RFC 2002, IP Mobility Support, Standards Track, October 1996.
- [OSA] 3GPP, Open Service Access specification, <http://www.3gpp.org/TB/CN/CN5/CN5.htm>
- [Parlay] Parlay Consortium, <http://www.parlay.org>.
- [ParlayX] Parlay Consortium, Parlay X Web Services Specification, Version 1.0, www.parlay.org
- [Prasad00] N.R. Prasad, IEEE 802.11 system design, IEEE International Conference on Personal Wireless Communications, December 2000
- [Samuel03] Isaac Samuel, Kishore Arora, Bhuvarahamurthy Narasimhan, Location-based performance-measuring techniques in UMTS, Bell Labs Technical Journal, Published by Wiley Periodicals Inc., Volume 8, Issue 2 (2003), (p 15-32)
- [SOAP] World Wide Web Consortium, "Simple Object Access Protocol (SOAP)", version 1.1, W3C Note 8 May 2000, <<http://www.w3.org/TR/SOAP/>>.
- [UMTSPos] 3rd Generation Partnership Project, "Stage 2 functional specification of User Equipment (UE) positioning in UTRAN," 3GPP TS 25.305, Release 5, Sept 2003.
- [Zhao02] Yilin Zhao, Standardization of Mobile Phone Positioning for 3G Systems, IEEE Communications Magazine, July 2002.