

Supporting Proactive Location-Aware Services in Cellular Networks

Axel Küpper¹, Florian Fuchs², Michael Schiffers¹, and Thomas Buchholz¹

¹ Munich Network Management Team, University of Munich,
Munich, Germany

{kuepper,schiffers,buchholz}@informatik.uni-muenchen.de

² Department of Informatics, Munich University of Technology,
Munich, Germany
fuchs@cs.tum.edu

Abstract. Location-aware services (LAS) adapt content and service execution to the user's current physical location. In today's cellular networks, only reactive LASs are offered. The user has to explicitly request the service before the network determines her current location and starts service execution. However, in the future the user would like to see proactive LASs, which are automatically triggered as soon as she enters a predefined point of interest. In contrast to reactive LASs, this requires that the network performs a continuous user tracking, which entails expensive point-to-point signaling at the air interface.

However, proactive LASs can be supported more efficiently if the mobile terminal locates itself. The terminal can derive its own location by processing location information which is broadcasted by the network or by a dedicated localization infrastructure like GPS. The network does not need to know where the user is, unless she has entered a point of interest. This paper follows this idea and presents the Location-Aware Mobility Architecture (LAMA), a platform for deriving and processing location information at the mobile terminal.

1 Introduction

"Pst, you have 'Buy new album of Madonna' on your To-Do list. This album is on offer at the record store on your right. Don't you want to drop in?" This could be a notification generated by a so-called *location-aware service* (LAS), which is offered by a service provider and delivered by the operator of a cellular network like GSM or UMTS through a *mobile terminal* (MT). In contrast to conventional mobile services, LASs adapt service execution or content delivery to the user's current physical location and thus allow a new dimension in service personalization.

Today's LASs are offered as *Reactive LASs* (RLAS) only. Location-aware information is only passed on demand to the user. She has to know by herself or at least assume that there may be a *point of interest* (e.g., a record store) nearby before going online to establish a service session and to request the LAS in order to retrieve the desired information. However, from the point of view of the user,

Proactive LASs (PLAS) are desired, which are automatically initialized as soon as she enters a pre-defined point of interest (as described in the example above). This requires that the user fixes her points of interest prior to service usage in a *location profile* when subscribing for a PLAS. Such a location profile may be derived from the user's individual preferences, for example "interest in vinyl records". From these preferences, a PLAS service provider derives points of interest and assembles them in the user's location profile. In this example, this could be the locations of record stores offering rare vinyl records. When the user approaches one of her points of interest, she is automatically alarmed by the service provider and can seize the opportunity of passing by. While RLASs are widely extended in cellular networks in the meantime, PLASs have been barely realized so far, which is due to the technical limitations of positioning methods used in these network. Nevertheless, PLASs are increasingly gaining momentum in numerous research and standardization activities, see [1], [2], and [3] for example.

From this description, a more detailed comparison between RLASs and PLASs can be derived. RLASs are invoked by the user from a front-end application of her mobile device, for example a WAP browser. Upon service invocation, the network derives the user's location and service execution is triggered. However, PLASs usually must be triggered at service components located at the fixed-network site, e.g., at a WAP gateway, as soon as the user enters a point of interest. In contrast to RLASs, the network does not need to derive the user's location just once, but must locate the user continuously, which is called *user tracking* in the following. In addition, each location information derived in this way must be checked against the user's location profile, which is referred to as *location monitoring* in the following.

Obviously, user tracking is the crucial point when realizing PLASs. To keep the network informed about a user's location requires dedicated *point-to-point signaling* between the network and the MT. Using RLAS, the overhead associated with point-to-point signaling can be neglected, as the user needs to be located only once or twice. However, if executed on a regular basis and for a substantial amount of subscribers, the signaling traffic dramatically increases and significantly burdens the air interface. In [1], this problem is referred to as "the dilemma of cell-based proactive location-aware services". Furthermore, whenever the network obtains new location information, location monitoring must be activated, which causes further signaling and additionally occupies computational resources.

To efficiently support PLASs, we propose the *Location Aware Mobility Architecture* (LAMA). The basic idea behind LAMA is that the network does not need to know where the user is, unless she has reached one of her points of interest. This implies that the MT is always aware of its own location and that it knows the user's points of interest. If and only if the user enters a certain point of interest, LAMA then signals this event to associated service components at the fixed-network site and triggers service execution in this way. Thus, LAMA combines user tracking with location monitoring at the MT. For an efficient user tracking, LAMA utilizes localization by *broadcast signaling*,

where location information is broadcasted in each cell of the network. From this information, LAMA can derive the current location and thereby avoids dedicated point-to-point signaling between the MT and the network.

The remainder of this paper is organized as follows. The subsequent section gives an overview of available technologies for localization, which are called *positioning methods*, and discusses their pros and cons with regard to PLASs. Section 3 introduces the system design of LAMA, i.e., its components and the interactions between them. Section 4 describes our LAMA prototype realization and demonstrates the deployment of LAMA by means of an application scenario. LAMA has been evaluated in a field test, which is described in Sect. 5. Finally, Sect. 6 briefly covers related work, and Sect. 7 gives an outlook on further work.

2 Positioning Methods

A positioning method comprises the infrastructure, physical fundamentals (i.e., the propagation delay of signals), and the technical process for deriving a user's location. Each tracking process applies a certain positioning method and fixes its parameters, such as the frequency of positioning and the user it is applied for. Cellular systems offer various alternatives for positioning, see [4] and [5] for example, but not all of them are applicable for PLASs.

Table 1 gives an overview of positioning methods applied in GSM and UMTS networks and lists some of their features. It is distinguished between *network* and *terminal-based positioning*, which denotes whether the location is derived at the network or at the MT. A further classification distinguishes in which mode of operation positioning is done. If it is executed in the context of an existing or forthcoming connection between the MT and the network, i.e., if the MT is busy, it is referred to as *online positioning*. If the MT is merely registered with the network, but no connection exists, i.e., if the MT is idle, it is called *offline positioning*. Note that only offline positioning is applicable for PLASs, because a connection is not established before the user has entered one of her pre-defined locations.

Furthermore, Tab. 1 shows for each positioning method the kind of signaling (broadcast vs. point-to-point), the accuracy, and the format location information is delivered in. A location format describes how location is represented and is usually based on a certain reference system. If this reference system is a coordinate system, location is available in a *physical format*. Typical examples are the *World Geodetic System 1984* (WGS84), *Universal Transversal Mercator* (UTM) and *Gauß-Krüger* (GK) coordinate systems. A *symbolic format* describes an object's location by abstract names or identifiers. In cellular networks, usual symbolic formats are *cell* and *location-area identifiers* (CIs and LAIs).

As can be derived from this table, some methods are based on broadcast signaling, for example LAI, CI, and GK-broadcast positioning. LAI-broadcast refers to the partitioning of an operator's coverage area into *location areas*, each represented by an unambiguous LAI. A location area comprises a sufficient number of cells and each cell, in turn, is assigned another identifier, the CI. Each

Table 1. Overview of positioning methods and their features

Positioning method	Net-work	Ter-minal	On-line	Off-line	Signaling	Accuracy	Format	Appl. for PLAS
LAI-broadcast		×	×	×	broadc.	5–15km	symbolic	yes
LA-update	×		×	×	p2p	5–15km	symbolic	yes
Paging	×			×	p2p	500m–10km	symbolic	no
CI-broadcast		×	×	×	broadc.	500m–10km	symbolic	yes
Cell-update	×		×		p2p	500m–10km	phy./symb.	no
GK-broadcast		×	×	×	broadc.	500m–10km	physical	yes
TA	×	×	×		p2p	10m–10km	phys./symb.	no
TOA	×		×		p2p	20m	physical	no
E-OTD		×	×	×	broadc.	20m	physical	yes
GPS		×	×	×	broadc.	20m	physical	yes

base station permanently broadcasts the LAI and CI of its location area and cell on a signaling channel. A MT is permanently listening to these channels, be it in idle or in busy mode, and is therefore always aware of these identifiers. Unfortunately, LAI and CI are symbolic formats, which is not very convenient for LASs. To overcome this lack, base stations can additionally broadcast location information in a physical format. For example, this is done in the German o2 network, where each base station transmits its own GK-coordinates using the SMS cell broadcast service [6].

To enable the network to locate the user, dedicated point-to-point signaling is required. This operation is called *location update* and is initialized each time the MT recognizes that either the LAI or the CI has changed. In the former case, it is called *location-area update* (LA update), in the latter *cell update*. Cellular networks only apply LA updates, which cause a moderate signaling overhead in contrast to cell updates. Because cell updates are not applied, the network must page the user's MT to determine her current cell. *Paging* is another positioning method that is based on point-to-point signaling. Like LA-updates, it is executed in the context of location management if a mobile-terminated call arrives for a user. For a deeper introduction into location management of cellular networks, see [7] and [8].

The positioning methods presented so far show only a moderate accuracy, which is not adequate for many LASs. Higher accuracy can be achieved if CI-broadcast or cell-update positioning are combined with the *timing advance* (TA) parameter, which refers to the distance between an MT and the serving base station. It is actually needed for synchronizing the TDMA medium access between MTs located at different distances to the serving base station, but can also be utilized for refining the accuracy of positioning. The distance is measured by means of the signal propagation delay and transmitted from the serving base station to the MT. It requires a dedicated point-to-point signaling and is only applicable for online-positioning.

A further improvement of accuracy can be achieved with triangulation. It is applied in the *Global Positioning System* (GPS), where the current location is

derived by signal propagation delays between at least three satellites and a GPS receiver. The principle of triangulation can also be realized by the cellular network, where the signal propagation delays to or from at least three neighboring base stations are processed. The triangulation method is called *Time of Arrival* (TOA) if the measurements and calculations for determining the location are done in the uplink, i.e., by the network. If it is done in the downlink by the MT, it is called *Enhanced Observed Time Difference* (E-OTD). TOA is only applicable if the MT is busy and requires point-to-point signaling. This is not necessary if GPS and E-OTD are used, because all parameters required to calculate the location are broadcasted in the downlink.

To draw a conclusion, user tracking should be supported by positioning methods based on broadcast signaling. This kind of signaling easily enables a MT to locate itself and, in contrast to point-to-point signaling, is independent of the number of subscribers for which user tracking is performed. Furthermore, some of the broadcast positioning methods are needed for location management anyway and therefore cause no additional overhead. As a consequence, possible candidates for user tracking in LAMA are LAI, CI, and GK-broadcast positioning as well as E-OTD and GPS. However, it must be stressed that the power consumption of MTs increases with positioning accuracy. For example, E-OTD and GPS are based on complex calculations for triangulation, which significantly exhaust the batteries of MTs, especially if they are continuously performed for user tracking. Therefore, E-OTD and GPS should only be deployed if high accuracy is requested by the associated PLAS.

3 The LAMA System Design

LAMA is an integrated platform which combines user tracking and location monitoring at the MT. Because of this integration and because LAMA preferably deploys positioning methods which are based on broadcast signaling, LAMA significantly reduces signaling traffic in comparison to approaches where user tracking and location monitoring are performed at the fixed-network site. If location monitoring has detected that the user has entered a point of interest, LAMA notifies an associated PLAS service component about this event and triggers service execution in this way. This service component can either be located at the MT or at the fixed-network site.

LAMA has been designed with the following design goals in mind:

1. *Configurability*: This design goal primarily focuses on the set of positioning methods installed at a LAMA-enabled MT. The availability of a certain positioning method depends on the capabilities of the MT, the underlying wireless system, and the operator the user is registered with. As a consequence, it is desired that LAMA, i.e., the set of positioning methods used by LAMA, can be tailored on an individual basis according to this environment.
2. *Transparency*: The tracking process should be transparent to both the service developer and the user. The service developer should easily and rapidly create new services, without taking care for positioning methods supported by the

respective operator, system, or MT. On the other hand, the user should not be concerned with aspects of tracking once she had passed through the necessary configurations of the system, e.g., determination of her preferred positioning methods. This makes the system easy to use.

3. *Generality*: The interfaces for accessing LAMA should be very general so that it is not restricted to specific PLASs, but can be used universally.
4. *Portability*: LAMA should be portable in order to be used in conjunction with mobiles of different types, e.g., mobile phones, PDAs, or a combination of both.
5. *Extensibility*: Although LAMA has initially been designed for operation in second and third generation cellular networks, it should also cope with proprietary and standardized positioning methods used in present and future wireless systems. This is very important, because it is expected that cellular technologies like UMTS will converge with local wireless systems like IEEE 802.11. Therefore, LAMA must be easily extensible in order to support converging networks.

Following these design goals, LAMA has been decomposed into three subsystems: the *location monitor*, the *location tracker*, and the *location converter*. The location tracker performs the user tracking and hides the associated technical details of the tracking process and the used positioning methods from the other subsystems. It determines the MT's location by using one or several positioning methods and delivers it to the location monitor.

The location tracker consists of one or more *trackers* and a central *tracker controller* which coordinates the different trackers (see Fig. 1a). Other subsystems access the location tracker only via the tracker controller. A tracker implements a certain positioning method and delivers the derived location to the tracker controller. In addition, it associates location information with quality parameters, such as accuracy (average error) and up-to-dateness (time stamp). LAMA can adopt any terminal-based positioning method and provides the corresponding trackers. In correspondence to Tab. 1, Fig. 1a shows trackers for LAI-broadcast, CI-broadcast, GK-broadcast, TA, E-OTD, and GPS positioning. If other positioning methods are supported by the underlying mobile environment, LAMA can be easily extended by appropriate trackers.

The life cycles of trackers are coordinated by the tracker controller. This component initializes and terminates trackers and processes the location information they deliver. To access different types of trackers in a unified way, each tracker has to implement the general **Tracker** interface. This interface provides operations to start and stop the tracking process, which are invoked by the tracker controller and which are implemented in the trackers according to the special hardware and software of the positioning method they are based on. Every time one of the trackers measures a new location, it notifies the tracker controller by calling its `locationTracked()` method.

From the locations received in this way, the tracker controller selects the location with the highest accuracy and up-to-dateness and delivers it to the location monitor. The location monitor observes a user's location received by the location

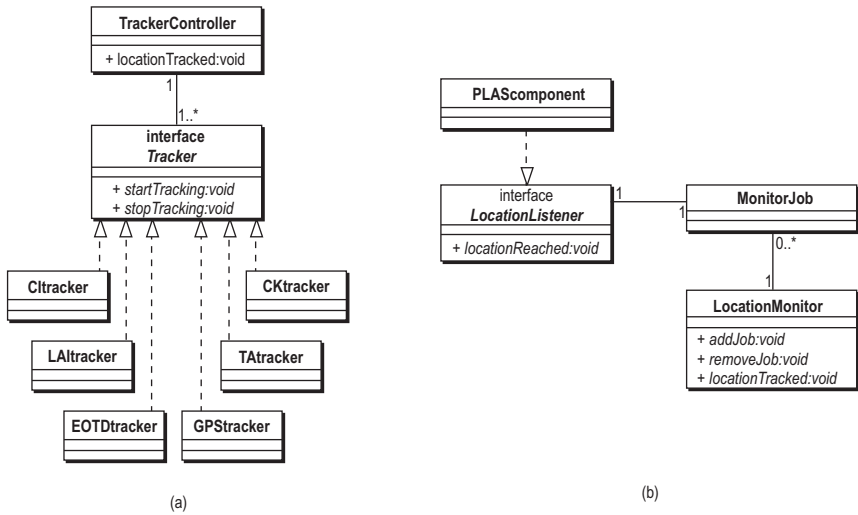


Fig. 1. Object design of the LocationTracker subsystem

tracker on behalf of a PLAS and notifies the associated PLAS service component as soon as a certain location has been entered. The object design of the location monitor is depicted in Fig. 1b. The location monitor manages several monitor jobs, each created by a PLAS service component and each containing one or several points of interest of a user who has subscribed for that PLAS. In order to be notified when the current location matches with one of the points of interest, the calling PLAS component must implement the `LocationListener` interface. Location listeners add new monitor jobs to the location monitor by calling the `addJob()` method and they remove jobs by calling the `removeJob()` method. Every time the location monitor receives updated location information from the location tracker, all monitor jobs are successively activated and executed in order to check the received location against the predefined points of interest. In case of a matching, the location listener of the associated PLAS component is notified that a certain point of interest has been entered. This is done by invoking the `locationReached()` method of the `LocationListener` interface.

The location converter allows to convert location information between different formats, for example from the GK to the WGS84 coordinate system and vice versa. It is used by the location tracker if a tracker does not support a certain physical or symbolic format as required by the invoking LAS. The location converter can also be used by the location monitor if it is necessary to convert a location profile provided by a PLAS. If a transformation between physical and symbolic formats is needed, the location converter at the MT must be connected to a remote directory. For example, if a PLAS requires the location in GK-coordinates and the CI-tracker, delivering locations as CIs, is the only available tracker, then the associated GK-coordinates must be obtained from the remote directory, which performs the mapping between these formats. In order to limit the number of such remote invocations, intelligent caching mechanisms are needed which buffer multiple mappings at the MT.

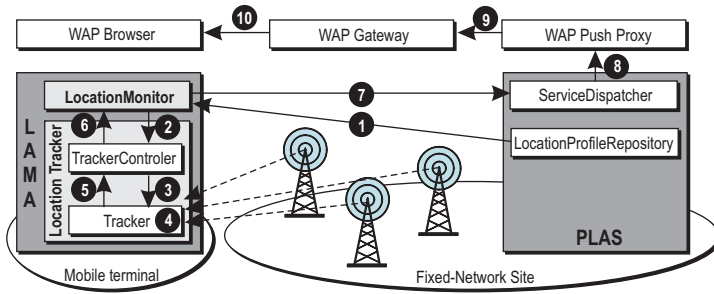


Fig. 2. Interactions between LAMA and a WAP service environment

Figure 2 shows how these LAMA components interact with PLAS components located at the fixed-network site (the location converter is disregarded in this example). The *Wireless Application Protocol* (WAP) has been chosen as the underlying service platform, which consists of a WAP browser at the MT as well as a WAP gateway and a push proxy at the fixed-network site. Generally, LAMA can be deployed independently of any service platform. For example, the WAP components can be replaced by an SMS infrastructure or by toolkits intended for UMTS. Furthermore, for the interactions between the MT and the fixed-network site the *General Packet Radio Service* (GPRS) is used.

In this scenario it is assumed that a location profile has already been generated for a particular user and stored in a location profile repository at the fixed-network site. This repository invokes the `addJob()` method at the location monitor with the location profile and the `LocationListener` interface of the service dispatcher as parameters (1). The location monitor activates the tracker controller (2), which, in turn, starts one or several trackers (3). Each time a tracker derives a new location (4), it returns it to the tracker controller (5), which forwards it to the location monitor (6). The location monitor then checks the location against the user's location profile. In case of a matching, it invokes the `LocationListener` interface of the service dispatcher at the fixed-network site and transfers the point of interest the user has reached (7). The next steps are not LAMA specific, but depend on the used service platform. The dispatcher might dynamically generate location-dependent content and combine it with a specific URL. This URL is then forwarded over a push proxy (WAP uses the term *push* instead of *proactive*) (8) and a WAP gateway (9) to the user's WAP browser (10), which can then request this URL.

4 Prototype Realization

Generally, today's MTs are characterized by limited computational resources, marginal storage, and proprietary interfaces. To evaluate LAMA according to usability, configurability, and portability, a prototype has been realized. The hardware configuration, i.e., the MT, consists of a Palm Vx handheld and a Siemens S35i mobile phone, which are connected via the serial interface. It is of course desired to use a single device with integrated radio facilities, why

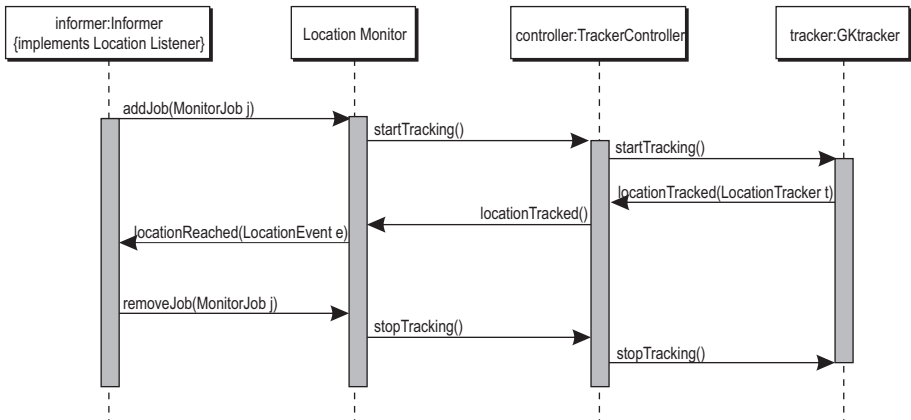


Fig. 3. Sequence diagram of the interaction between the LAMA components and Informer

another version of LAMA for the IPAQ handheld is in preparation as well. We have focused on GK-broadcast and GPS positioning and implemented the corresponding trackers. This configuration has been deployed in the German o2 network, where GK-broadcast is available. The mobile phone is used to receive GK-broadcast messages and to interact with LAMA and PLAS components located at the fixed-network site. GK-broadcast messages are requested from the handheld over the serial interface using AT commands according to [9]. Communication with remote LAMA and PLAS components can be done over SMS or GPRS bearer services.

To deploy LAMA at different types of MTs, we have focused on a Java-based solution for the software configuration. Basically, the *Java 2 Micro Edition* (J2ME) seems to be an appropriate platform, but suffers from the missing of some important device-dependent functions. Therefore, we have chosen the WebSphere Studio Developer (WSDD) platform from IBM [10]. WSDD is nearly J2ME-compliant, offers the desired functions for different hardware configurations, and is available for most of the popular handheld operating systems.

To easily test LAMA, an application called *Informer* has been implemented as well. Informer notifies the user about landmarks in his close proximity and, if desired, provides additional information about these landmarks. A landmark consists of the location in GK-coordinates, the name of the point of interest, and a short description. For example: 3690021, 5341076, “Munich Olympic Tower”, “Landmark of the Olympic Summer Games 1972 in Munich”. A landmark can be set at the current location. Alternatively, a landmark location profile, which could be, for example, offered by a service provider for selected cities, can be downloaded from a remote server. Informer is realized as a J2ME application and imports the LAMA classes. Informer is an application for *Personal Information Management* (PIM), like calendars and to-do lists, and is primarily executed at the MT. It interacts with the LAMA system in the following way (see figure 3):

1. Informer implements the interface `LocationListener` consisting of the public method `void locationReached(LocationEvent e)`. This method will be called by LAMA every time the location monitor recognizes a matching of the current location with a point of interest.
2. Informer creates a new job for the location monitor by creating a monitor job with a reference to the corresponding object implementing the `LocationListener` interface. In a subsequent step, it adds the job to the current location monitor. The tracking process is automatically started by LAMA when the first job is added to the location monitor. Tracking is automatically stopped, when there is no more job.
3. A job has to be removed from the location monitor by the service component which has initialized this job. This is done by calling the method `void removeJob(MonitorJob j)`.

5 Field Test Evaluation

To evaluate the LAMA concept in general and the accuracy of the GK-tracker in particular, a field test including 81 radio cells of the o2 network in Munich has been carried out (see Fig. 4). For this purpose, the sample application Informer has been arranged with a location profile containing popular landmarks of Munich.

During the field test, LAMA worked with a high reliability and displayed neighboring landmarks with an acceptable accuracy (when used in conjunction with the GK-tracker). In addition to the GK tracker, a GPS tracker has been implemented and both of them have been deployed in parallel. Due to simplicity, we have decided to choose GK-coordinates as our reference format, making it necessary to translate the WGS84 coordinates (delivered by the GPS tracker) using the location converter. From an economical point of view, a further advantage is that third party providers can offer PLASs without the need to enter into cooperation with network operators. However, this is in contrast to the efforts of operators, who probably want to gain revenues by selling location information.

LAMA has only been tested with the landmarks of Munich, i.e., with a manageable number of points of interest. However, MTs are equipped with limited storage and processing capabilities, which is why scalability problems have to be expected if the number of points of interest to be checked by the location monitor increases a certain order of magnitude. For example, confronting the location monitor with all landmarks of Germany or even the entire world would overload the system and would be inefficient anyway, because the users' mobility patterns are dominated by local movements. It is therefore obvious to charge the location monitor only with points of interest of local relevance and to automatically refresh the monitor jobs in dependence on the user's current location (see also Sect. 7).

To evaluate the accuracy of GKtracker the following data has been logged during the field test : GK-coordinates sent out by the base stations (*base-station coordinates*) and the actual coordinates of the MT when new GK coordinates

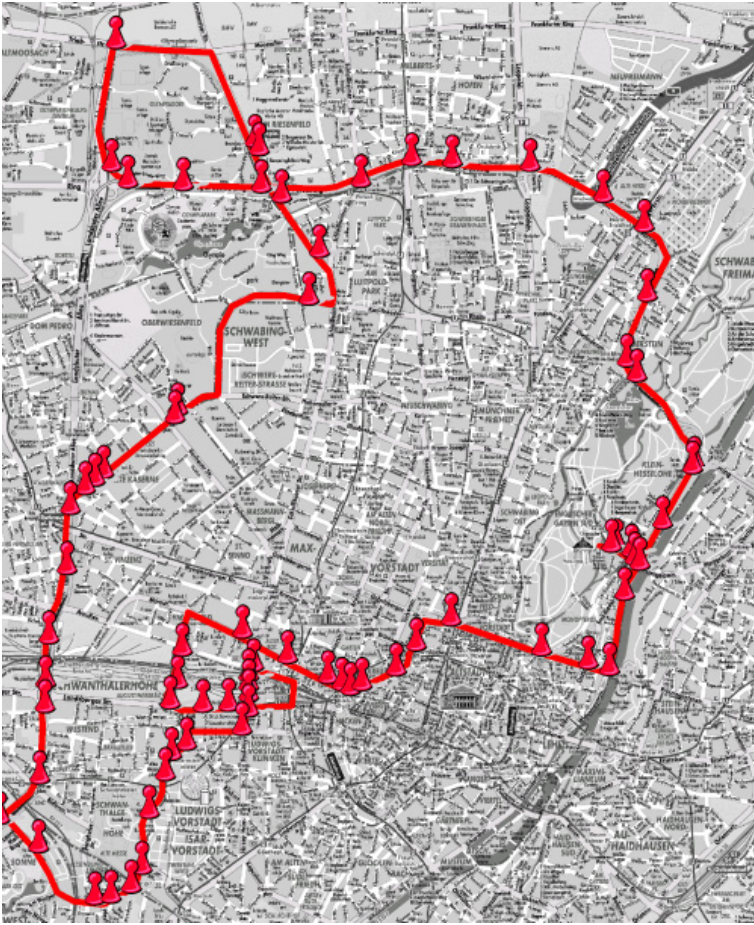


Fig. 4. Geographic overview of the field test

arrive (*arrival coordinates*). The arrival of new GK coordinates is referred to as *location update* in the following. The accuracy of the GK-tracker has been determined by calculating the *positioning error*. This error refers to the distance in meters between the arrival coordinates of a location update and its associated base-station coordinates. Each cell has been entered for several times from different directions. Correspondingly, *maximum*(MaxPE), *minimum* (MinPE), and *average positioning errors* (APE) could be derived for each of the 81 cells. In addition, the *location-update distance* (LUD) has been evaluated, i.e., the distance in meters between the arrival of two consecutive location updates.

Figure 5 shows the area around Munich Central Station, a clipping of the entire field test, and the measured base station and arrival coordinates. Table 2 contains the associated values for MaxPE, MinPE, APE, and LUD. The average MaxPE of all cells covered by the field test amounts to 595 meters, whereas the average APE is 429 meters. The smallest MinPE that has been derived is

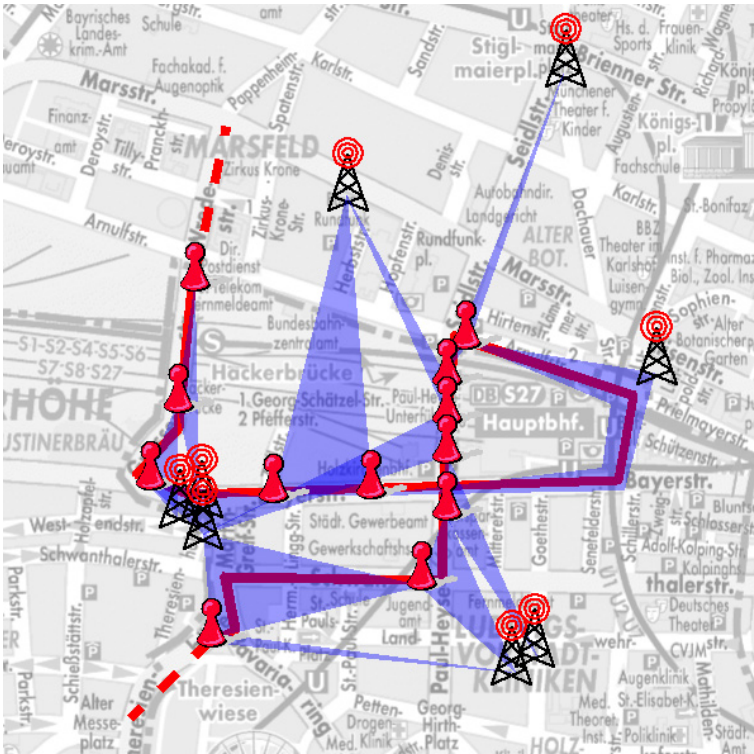


Fig. 5. Graphical evaluation showing the area around Munich Central Station

98 meters. The average LUD amounts to 345 meters. Obviously, the positioning error depends on the density of base stations in a certain area. Usually, this density is much higher in urban areas than in suburban or rural areas. However, the results achieved by the field test can be considered to be representative for cellular network in urban areas.

To draw a conclusion, the field test has demonstrated that a terminal-based approach of user tracking in conjunction with location monitoring is a promising way to realize PLASSs. Currently, this approach is further refined in that trackers installed at an MT are coordinated in an adaptive way in dependence on the user's current distance from a point of interest.

6 Related Work

One of the pioneers in location awareness is the *Active Badge* system which has been developed in 1992 [11]. It enables the localization of individuals in buildings by determining the location of their Active Badge. An Active Badge sends out an infrared beacon every 10 seconds which is detected by stationary sensors installed within a building. Thus, one can say that Active Badge realizes network-based positioning. Principally, this approach can be adopted to realize

Table 2. Measured distances corresponding to the area shown in figure 5

Cell	MinPE	MaxPE	APE	LUD
50	641	641	641,0	0
51	223	469	346,0	541
52	262	355	308,5	183
53	345	451	398,0	118
54	541	578	559,5	86
55	423	500	461,5	95
56	596	670	633,0	76
57	400	641	520,5	1033
58	618	646	632,0	204
59	149	165	157,0	250
60	100	231	165,5	200
61	206	434	320,0	255

PLASs. However, its implementation in cellular networks would cause a significant signaling overhead if MTs of a substantial amount of subscribers would frequently send out these beacons.

The *Guide* project has focused on another application and developed a tour guide for the city of Lancaster [12]. Unlike Active Badge, beacons containing location information are sent out by the network and received by the terminal. The system has been built upon an IEEE 802.11 infrastructure, why the relevant places to be covered by the tour guide have to be equipped with IEEE 802.11 access points. Apart from the limitation that Guide focuses on a certain positioning method, its designers have restricted to reactive services only.

As interactive data-oriented services for cellular networks have gained more and more attraction in the recent years, an increasing number of projects have started making services in these networks location-aware. The *Deep Map* project has developed a system for GSM to exchange location information between members of a tourist group [13]. The system is based on traces which contain the coordinates of representative tours through the area to be covered by the service. These traces have to be fixed in advance and stored on a central server. The current location of a group member can then be predicted by the server. In order to obtain the initial location or to correct the prediction, the terminal needs to perform location updates, i.e., the current location is measured by GPS and transferred to the server. However, the number of these location updates can be dramatically reduced according to a system without location prediction. Although this approach can also be adopted for PLASs, there is too much overhead for the acquisition of representative tours, especially for operators serving a national-wide coverage area.

One of the first platforms that combines different positioning methods has been developed by the *Nexus* project [2]. Nexus proposes an integrated platform which incorporates different positioning methods for indoor and outdoor positioning as well as a spatial data model for representing the physical world. Thus, Nexus is very close to the approach followed by LAMA. The *Lol@* project has

developed another platform for location-aware services [3]. Unlike the aforementioned platforms, Lol@ explicitly addresses PLASs. It defines a network-centric *Gateway Mobile Location Center* (GMLC) which performs the user tracking and triggers content delivery as soon as the user enters a point of interest. The GMLC is assisted by positioning methods located either inside the MT or at the network. However, location monitoring is executed by the GMLC at the fixed-network site and thus, dedicated point-to-point signaling between MTs and the GMLC is needed to support PLASs.

7 Outlook

Generally, the costs of user tracking increase with the accuracy of the underlying positioning method. It is therefore intended to add more intelligence to LAMA. Positioning methods should be automatically activated and deactivated in dependence on their costs, accuracy requirements of the used PLAS, and the user's current distance from a point of interest. For example, it would be desired to switch from a positioning method with moderate accuracy to one with higher accuracy if the user approaches a point of interest and if this degree of accuracy is really required for the associated PLAS. LAMA is currently extended to provide these functions.

Furthermore, ongoing work concentrates on an extension of the location monitor in order to avoid scalability problems at the MT, which occur if too many points of interest have to be checked against the current location. It is intended that the monitor jobs only contain a subset of points of interest that are located within a certain distance to the user's current position or within her city or region. Thus, the location monitor itself becomes location-aware, and monitor jobs must be automatically refreshed in dependence on the user's location.

Acknowledgment. The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments during the final phase of the paper. The MNM Team directed by Prof. Dr. H.-G. Hegering and Prof. Dr. C. Linnhoff-Popien is a group of researchers of the University of Munich, the Munich University of Technology, and the Leibniz Supercomputing Center of the Bavarian Academy of Science. For more information see <http://wwwmnmteam.informatik.uni-muenchen.de>.

References

1. Fischmeister, S., Menkhaus, G.: The Dilemma of Cell-based Proactive Location-Aware Services. Technical Report TR-C042, Software Research Lab, University of Constance (2002)
2. Klinec, D., Volz, S.: NEXUS – Positioning and Communication Environment for Spatially Aware Applications. In: International Archives of Photogrammetry and Remote Sensing. Volume XXXIII of Inter-Technical Commission, IC 10., Amsterdam, Netherlands (2000)

3. Popischil, G., Stadler, J., Miladinovic, I.: A Location-based Push Architecture using SIP. In: Proceedings of the 4th International Symposium on Wireless Personal Multimedia Communications (WPMC '01, Aalborg, Denmark (2001)
4. 3GPP TS 02.71: Location Services; Service Description, Stage 1 (2001)
5. 3GPP TS 03.71: Location Services; Functional Description, Stage 2 (2002)
6. 3GPP TS Group Terminals 03.41: Technical Realization of Cell Broadcast Service (CBS) (1998)
7. Tabbane, S.: Location Management Methods for Third-Generation Mobile Systems. *IEEE Communications Magazine* (1997) 72–84
8. Zhang, J.: Location Management in Cellular Networks. In: *Handbook of Wireless Networks and Mobile Computing*. John Wiley & Sons (2002) 27–49
9. 3GPP TS Group Terminals 07.07: AT Command Set for GSM Mobile Equipment (ME) (1998)
10. : (<http://www.ibm.com/embedded/>)
11. Want, R., Hopper, A., Falcao, V., Gibbons, J.: The Active Badge Location System. *ACM Transactions on Information Systems* (1992) 91–102
12. Davies, N., Cheverst, K., Mitchell, K., Efrat, A.: Using and Determining Location in a Context-Sensitive Tour Guide. *IEEE Computer* (2001) 35–41
13. Walther, U., Fischer, S.: Metropolitan Area Mobile Services to Support Virtual Groups. *IEEE Transactions on Mobile Computing* **1** (2003) 96–110