Towards adaptive location-aware mobile assistants

Jörg Baus * Yun Ding † Christian Kray ‡ Ulrich Walther §

ABSTRACT

In this paper we first look at a scenario to illustrate what services a truly adaptive mobile personal digital assistant should provide and how these should be offered to an untrained user. We identify relevant factors to be taken into account by a mobile assistance system and shortly review the current state of the art in mobile systems. We then propose an agent-based resource-adaptive framework to cope with the inherent complexity and dynamics of such systems. This framework does not only address important technical problems (such as networking issues, location detection, and resource constraints) but also provides a mobile assistant with several strategies to adapt its services to the current situation.

Alice in Tomorrowland

... burning with curiosity, she ran across the field after the strange rabbit and followed it into the rabbit-hole. At the sides of the well, she noticed that they were filled with cupboards and book-shelves. On one of them, she saw a small device, which lit as she passed: 'TAKE ME, I'M YOUR ASSISTANT' appeared on the tiny screen. So, she took it. Down, down, down. 'I wonder how many miles I've fallen by this time?' she said aloud. To her surprise the device in her hand answered: 'Oh, already tens of miles. In less than a minute, you will land safely'. Somewhat relieved now, Alice was glad as she indeed landed on some dry leafs shortly thereafter, and hurried past the rabbit. Then she came to big hall with doors all around it, and she didn't know which way to go. As she was wondering what to do next, her assistant suddenly spoke to her again: 'If you want to get out of this hall, there is a key on the glass table right over there. It opens the small door behind that low curtain. In order to get through, you need to drink ....'

slightly modified version of Lewis Carroll's *Alice's Adventures in Wonderland*

*University of Saarland, Computer Science Dept., D-66123, Saarbrücken, Germany, baus@cs.uni-sb.de

†European Media Laboratory, Schloss-Wolfsbrunnenweg 33, D-69118 Heidelberg, Germany, yun.ding@eml.villa-bosch.de

‡DFKI GmbH, Stuhlsatzenhausenweg 3, D-66123, Saarbrücken, Germany, kray@dfki.de

§International University, Campus 1, D-76646 Bruchsal, Germany, ulrich.walther@i-u.de

1. INTRODUCTION

The goal of a location-aware mobile assistance system is to assist users everywhere and at any time by providing the right information at the right place in the right way [9]. Typical services in this context include, for example, to navigate users through unfamiliar environments and to provide them with information related to their current or predicted future location. For instance, Alice's mobile assistant informed her of her immediate landing and called her attention to the key and the magic drink. This kind of information should be presented in a way, where the users' geographical location including their orientation (e.g. left to the castle) and the adjacency relationship to the sites (e.g. ten meters in front of the city hall), as well as their typically limited displaying, computing, communication and situational resources are taken into account. For example, The information about her immediate landing is given in short words to Alice because of the time pressure, while the advice on how to get out of the hall could be formulated in more detail. In order to fulfill this goal, we identified three main layers involved in a mobile assistance system:

1) In the *user layer*, location sensors gather location information of the users on move. In addition, information about the users (e.g. their interests and abilities) and the resources at their disposal (e.g. the available display device) are stored.

2) In the *service layer*, mobile devices (PDA, laptops or small computers embedded into some intelligent devices like wrist watches) offer services of different kinds.

3) The *network layer* enables the users to communicate with their environment by providing different kinds of communication infrastructures[1].

Characteristics of these layers define the constraints on mobile systems: Mobile resources such as in- and output devices (e.g. displays of mobile devices), computational resources (processor speed and memory capacity), and energy supply are rare compared to those in "static computing". While users change their locations, their connection to the environment must switch between the different communication infrastructures, which are highly variable in their availability, performance, reliability and cost. In order to provide mobile users with services of acceptable quality, agile *resource adaptation* is a key feature.

[1] Although communication resources are a kind of resources provided to the users, they will be separately handled due to their general importance and their technical particularities.

The remainder of this paper is organized as follows. We first examine the relevant factors for location-aware mobile assistants in order to derive the requirements on such systems (section 2). We then give an overview of the technical state of the art in recent mobile systems (section 3). In section 4 we propose for a layered agent-based adaptive framework for location-aware mobile assistance systems, and close on some concluding remarks in section 5.

2. RELEVANT FACTORS

A mobile assistant that knows about its location can provide users with information and services that are better tailored to the current situation, as it can filter its output according to the users' current position. While the *location* is a key factor to be taken into account, it is not sufficient by itself. Other important spatial features include, e. g. the direction the user is looking or heading, the travel speed, or the visibility of relevant objects. These should also be considered when accessing or presenting information about the environment; for example, a nearby building that is not directly visible or behind a user must be introduced differently than one that is directly in front of the user. However, positional information such as the location of the user and the features described above cannot be taken for granted: its recency, precision, and availability can vary over time and space. This must also be considered by a truly adaptive system.

However, there are further relevant factors that are not spatial per se but that are *related to the users' locations*. These include the availability, bandwidth, cost, and reliability of the current *network* connection, which depends on the location of the user and is especially important in case that a mobile device does rely on external data and services in order to provide assistance to the user. Additionally, the state of the *environment* is an important factor, which includes (but is not limited to) objects in the user's vicinity (e. g. a shop that is open or closed), the weather, and traffic conditions.

While the factors presented so far are related to the current location, there are also others that are related to the users and the resources at their disposition: A *user model* that stores interests and properties of a user can also greatly contribute to a truly adaptive system: It can help to select objects for reference among the usually large number of potential candidates, and it facilitates proactive behavior such as letting the user know when an interest site is nearby. Furthermore, knowledge about the user's preferences and abilities enables a system to tailor the presentation specifically to the user's needs and desires, e. g. relying more on audio or on graphics for presentations[1]. Information about the user's intentions and goals allows for additional improvements in adaptation; for example, the system might refrain from proactively informing the user about points of interest in his vicinity if his current goal is to get to the train station as quickly as possible. *Technical resources* such as computing power or output devices have a similar impact on how an adaptive assistant can provide its services. Finally, the user interface offers some further means of tailoring presentations to the users' situation, e.g. choosing among purely textual output, animated interface agents, audio output, etc.

3. STATE OF THE ART

This section surveys the state of the art of techniques used for recognition of the users' location, and reviews the issues that network infrastructures for mobile environment are facing.

3.1 Recognition of users' location

For the key factor in adaptive location-aware systems – the user's position – several different sensor devices are available that can be built into the mobile device. There is, for instance, the well-known GPS devices, or indoor infra-red location detection systems, active badge or passive tag systems, or magnetic compasses and barometric altimeters. All the input from these sensors has to be treated by the system in a well-defined way so that higher layers can access location information in a uniform way. For this purpose, the measured values from all the available sensors are merged into a consistent location state within the process of *sensor fusion*. Since multiple sensors can perceive possibly contradictory facts about the physical world, this information has to be combined to support reliable location information. At the same time, *sensor filtering* allows for detecting and ignoring gross measurement errors.

If the system knows about the exact position and orientation of the user it can provide reorientation instruction in a very simple manner, e.g., by means of an arrow. If in contrast the current position and orientation are vague or missing, the system must provide information to the users to locate themselves and to determine their actual orientation in space. Alternatively, the assistant can design the user dialog in a way that helps the user to fill in the missing information. In a scenario where the quality of position and orientation information is known beforehand (e.g., inside buildings equipped with infrared transmitters), the digital assistant can just passively present the information (prepared in a location sensitive manner) that it receives from the transmitters. This is what we call *passive location sensitivity*. In an outdoor scenario with varying GPS signal quality, the decision would have to be made by the device itself: The assistant has to select and generate the information depending on the actual position actively. This is what we call *active location sensitivity* Different types of situations with respect to the quality of position and orientation can be distinguished:

1. Sufficient position and orientation information
2. Sufficient position but insufficient orientation
3. Insufficient position and orientation information

There are also some interesting intermediate states, such as good orientation and mean position information or rough position but no orientation information.

In mobile environments, it is quite interesting and important to know if a user resides in a set of given regions, or if a user stands still for more than a certain period of time, which may indicate his interest in objects located at that position. In general, changes of users' locations can be polled or subscribed to using a *distance-based* or a *time-based/periodic* mechanism [10]. In the first case, the new location will be reported when a given distance threshold is reached, while in the second case, it is reported in a periodic manner, or at a specific point of time. These mechanisms can be used to realize the following two types of triggers to propagate location information:

- a *region-based trigger* is released, whenever the user enters or leaves a polygonal region, and
- a *time-based trigger* is released, if the user resides within a given distance threshold for longer than a specifiable amount of time.

Since the mobile users will mostly be in areas that are only covered by GSM networks, we must take into account the 9.6 kbit/s bandwidth limit, of which we only want to use a small fraction for location reporting to fulfill the location multicast to the group members. In order to minimize the number of location update messages on the wireless network, *location prediction* based on the current location state (e. g. position, direction and speed be seen as it minimum attributes, etc.) can be used.

3.2 Network infrastructure

In the context of mobile applications, most of the application services are provided on servers that have connection to the wireless devices carried by the user. Mobile networks, however, impose a new family of possible failure cases. These are mainly occasional disconnections from a network due to uncovered areas, very limited available bandwidth, and, in the case of multiple overlapping networks, the problem that one has to roam between networks with different IP addressing and topology.

The mobile devices may provide the base services to the users at a lower quality than the servers, so in case of a complete disconnection from all networks rudimentary services can be at least offered. But in general we assume that at least a GSM connection [7, 5] at 9.6 kbit/s will be available for the wireless devices most of the time. Additional wireless LAN (WLAN) connections with up to 11 Mbit/s may be available in special areas (the 'hot spots' that are visited by 90% of the users), but these resources are shared among all users connected to the same base station; therefore, the bandwidth available for one client might vary from a few kbit/s up to several hundreds of kbit/s. In these places with enhanced network connection, additional functionality like e.g. better voice quality or video streaming services can be offered. The required monitoring and signalling functionality that is needed to adapt to changing network conditions is described in section 4.2.

Since we assume that providers might be independent of each other (at least the GSM provider is different from the WLAN providers), the IP addressing scheme within the WLANs and the GSM network is completely different. In order to be able to support transparent network connectivity in all areas, mobile IP [11] has been introduced to support roaming and hand-off on the network layer: The mobile terminals are run in *colocated care-of-address mode*, that means when entering a new IP subnet the mobile terminal obtains a topologically correct IP address and registers itself with its home agent which tunnels the packets destined for the mobile terminal.

4. AN ADAPTIVE FRAMEWORK

In order to address the requirements specified in section 2, a system must be able to cope with a high level of complexity and dynamics, which are inherent for each layer of a mobile assistance system itself and the interactions among them. Based on an adaptive agent infrastructure (section 4.1), we present several ideas on how to adapt to varying quality of the network connections and location information (section 4.2). Then, several adaptation strategies of assistance services are presented in section 4.3, which are build on top these two layers, but also interact with them.

4.1 Agent infrastructure

As described above, each layer of a mobile assistant system pursues its own objectives, and interacts with each other in order to adapt itself rapidly to the changing environment. Due to this proactiveness, communication ability and reactivity, each layer can be modeled as *software agents* in a multi-agent system (MAS)[13]. We designed and implemented RAJA, a FIPA [6] compliant Resource-Adaptive Java Agent Infrastructure, for the development of resource-adaptive multi-agent systems [4]. In this context, the term *resource* has a broader meaning: it includes not only technical resources such as bandwidth and computing power, but also situational resources determined by the users' location and their environment, as well as user-oriented resources (cf. user-model in section 2). The architecture of RAJA is illustrated in figure 1.

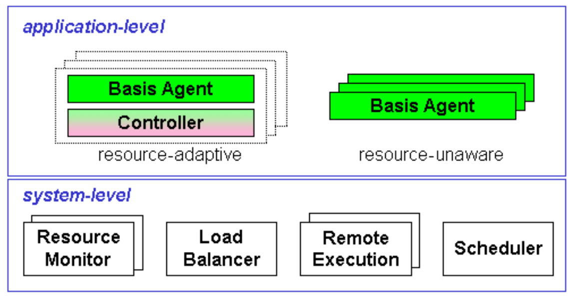

Figure 1: A reflective multi-level architecture

At the application-level, *basis agents* model domain-specific application functionality. At the system-level *meta agents* are part of the agent infrastructure. They provide system-level services addressing *resource management* to basis agents as well as to other meta agents. A Resource Monitor, for instance, records the availability of local resources, which range from CPU power and network bandwidth to I/O facilities such as displays and printers. A Load Balancer acquires load information of the basis agents and decides where to perform a pending task in order to minimize its execution time or to maximize the overall throughput. The decoupling of basis agents and meta agents separates non-functional resource management from application functionality. It reduces the complexity of single agents on one hand, and on the other hand, it allows the basis agents to share the functionalities provided by the meta agents. Meta agents are not allowed to directly examine and modify the state of the basis agents; this can only be done indirectly through *controllers*. A controller is only attached to a resource-adaptive basis agent; a resource-unaware basis agent does not have a controller. A controller performs reflective computation about the domain-specific computation of its associated basis agent. Examples of reflective computations are monitoring of performance, (e.g., execution time or result quality), monitoring of resource consumption, and most importantly, computation of adaptation decisions (e.g., which parameter or configuration should the application computation run with next and where should it run). The distinction between basis agents and their controllers eases the transformation of resource-unaware legacy systems into resource-adaptive systems by simply attaching controllers.

The following two sections describe the adaptation strategies applied in the network and the service layer respectively and how they can be realized using the RAJA agent infrastructure.

4.2 Adaptive networking

In order to support adaptivity in terms of network quality of service (QoS), and with *ad-hoc routing functionality* between the mobile terminals to enlarge the area with WLAN access, we need to extend the networking layer with *network monitoring*. While the current IP address (the currently used IP subnet) can be hidden from the application by using a mobile IP layer, the change in available bandwidth or latency offered by the different providers cannot. Therefore, the application has to be informed of the changes in order to adapt to the new conditions and to provide the user with the best possible service.

For this reason there is a RAJA basis agent, the Monitoring Agent, which is responsible for measuring the wireless link quality (e.g. using the Linux Wireless Extensions [12]) and for reporting available network access, bandwidth and predicted latency (currently supporting Ethernet, WaveLAN, and GSM) to the RAJA middleware, for instance, the meta agent Resource Monitor. In addition, the Monitoring Agent communicates with the underlying network monitor and informs the RAJA middleware of upcoming or completed handoffs, or temporary completed disconnections from the network. Application services interact with the Resource Monitor to adapt their services to meet the available network resources.

A further means of adaptation is provided by the idea of a mobile ad-hoc network: using each user's mobile device as an intermediate router to enhance the connectivity to wireless LAN networks. If the density of mobile devices (i. e. tourists) between WLANs is high enough, intermediate mobile stations can forward packets from senders that would otherwise be out of range. In these cases the mobile devices do not have to use the fallback to GSM/GPRS network, since they are still able to use the wireless LAN's higher bandwidth. In order to seamlessly integrate this ability, an *ad-hoc routing daemon* is situated right between the kernel IP layer and the RAJA middleware layer. It uses (mobile) IP to realize a virtual IP addressing scheme on top of the mobile IP layer. Even if the mobile user roams between networks utilizing different IP addresses, we are able to achieve consistent IP addressing. The ad-hoc routing functionality itself is realized through the use of these ad-hoc routing daemons that are running on all mobile devices. Each daemon can act as a router that forwards packets and uses discovery mechanisms in order to update its internal routing table when the connectivity to other daemons changes.

As a first implementation, we use the *Ad-hoc On Demand Distance Vector* routing algorithm (AODV) as specified in an internet draft by the IETF MANET group [2]. Currently, we are investigating existing ad-hoc routing protocols to identify features that are crucial for our application scenario. Since the mobile devices are equipped with GPS hardware, we will be able to use positional information as an additional metric for the routing process. Additional semantic information – like the underlying world model or knowledge about geographic and logical network topologies – might help to further improve the ad-hoc routing algorithm.

4.3 Adaptive services

There are several fundamental services such as information about the current environment, navigation, or support for localized services that a mobile assistant should provide in a way that is adapted to the current user and situation. In section 2, we argued that – aside from the current location – there are other important factors such as additional spatial features, user and context information, and knowledge about the state of the environment. How do these factors influence the provision of services?

Provision of information about the current environment may be the most fundamental service that can be expected of a mobile assistant. The first step in this process usually is to determine what object(s) the user wants to learn more about. While spatial information can help to restrict the set of candidates, it is rather likely that it is still too large, e. g., when there are several objects within sight of the user. In this case, knowledge from the user model and the dialog history can help to narrow the number of choices. For example, if the user is known to be interested in objects of a certain type or if an object was just introduced by the system, then a request for information about it is more likely. If the user is already familiar with an object, the probability of him asking to learn more about it is rather low. Using adaptation strategies like these will help the assistant to greatly reduce the number of times when it has to explicitly ask the user to select the object he wants to know more about. This can be crucial in cases where the user is in a hurry and does not want to engage into lengthy dialogs, but is also beneficial in general since the presentation of unnecessary information can be avoided. Moreover, it enables the system to act proactively by letting the user know when he is near some interesting site.

Once the target object of the actual query is known, information about it must be presented to the user. In this context, an adaptive assistant not only has to select the content but also the medium of the presentation. The depth and volume of the information can be adapted in several ways. On one hand, knowledge about the interests of the users and their current goal help to tailor the content to the users. On the other hand, the dialog history provides a means to refer to previously mentioned items and to infer what the user already knows and what he does not. The medium/media selected for the presentation also influences the content and vice versa, e. g., certain content can only be presented using a certain medium. Additionally, the choice of media depends on the user model (preferences, disabilities, etc.), the context model (e. g., visual attention distracted while driving a car), and the available resources (for example, low bandwidth does not allow for video streaming). The user interface design is also an important factor, which is discussed in section 4.4.

Navigation assistance is a second fundamental service in a mobile assistant. Most of the adaptation strategies presented above apply to navigation as well, since it includes the task of providing information about the environment, e. g., when introducing objects needed for navigational instructions. However, there are also some means of adaptation unique to the navigational service. The degree of detail of navigational instructions can be adjusted to the current situation using knowledge about the users' preferences, familiarity with the environment, and their current goal. For example, when they are in a hurry they probably prefer short instructions, but when they are on a sightseeing tour, more elaborate utterances, possibly highlighting interesting sites along their way, are more adequate. A central task in the context of navigation is the selection of reference objects (needed, e. g. to generate spatial prepositions such as "left of" [3]), which can greatly benefit from the above mentioned knowledge sources. Factors such as the users' interest, their familiarity with the environment, or whether an object has been previously mentioned all influence the quality of a potential reference object, that has been preselected using purely spatial criteria. In addition to tailoring the route instructions to the current situation, the route planning process itself can also be realized adaptively in a location-aware mobile assistant. This does not only concern the a priori route- or tour-planning: knowledge about users' current position (and possibly their current target/goal) allow for replanning to take place instantaneously when he deviates from a previously chosen route.

A third example for fundamental services to be provided by a mobile assistant is the support of adaptation to the current situation and location for other (external) services such as hotel reservation systems, public transportation schedules, etc. These services require a positional information of a certain granularity/precision[2], which a mobile assistant can deliver. However, if for some reason the information is not available/measurable at the desired quality, there are some means to adapt to this situation. On one hand, the assistant may use non-positional knowledge, e. g. a geographic information system or context model, to *infer* the requested information. On the other hand, it is possible to directly interact with the user to obtain missing information from him, which can then serve as a basis for the inference process.

In order to address the inherent complexity of a system that aims at providing such adaptive services, an agent-based approach seems to be well suited as it not only allows to encapsulate functionalities but also makes interactions explicit. For example, a spatial reasoning agent was implemented using a multi-agent system, which proved to be very beneficial. By realizing important subprocesses, e. g. the computation of spatial relations or good reference objects, as autonomous agents we were able to take into account many spatial and non-spatial factors [8].

These examples illustrate how different services can benefit from adaptation to the current location and situation, and by what means this can be achieved. While the user experience will certainly improve when these service level adaptations are realized, the user interface itself can also be adapted to make a mobile assistant more user-friendly.

4.4 Adaptive user interface

Knowledge about the current position does not only influence the way services can be provided by the systems, but also how those services are *presented* to the user. In case the system knows enough about the actual position and orientation of the user, it can produce a simple navigation instruction (e.g. an arrow as shown in figure 2A). If the quality of the orientation information decreases and the system cannot exactly tell where the user is looking at, a simple arrow could mislead the user. Therefore additional information about the choices at the decision point has to be provided. Figure 2B shows such a graphical way description for an orientation resolution of +/-90 degrees. The topological diagram includes only the different choices at the current decision point, but does not show any additional landmarks. Please note that the map can still be roughly aligned to the user's walking direction to simplify reorientation.

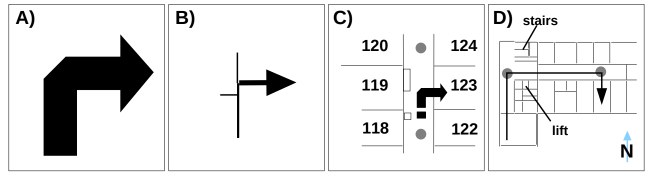

Figure 2: Four different graphical way description schemata in the passive location sensitivity mode

As the quality of orientation and the position information declines further landmarks have to be included. Figure 2C shows a description where the position resolution covers three potential decision points (two are indicated as grey dots and the third one is covered by the arrow). In such situations a purely topological map could cause problems and therefore an appropriately clipped area of the surrounding (here: the adjacent rooms with numbers and parts of the hallway, pillars and a locker) are displayed. By clicking on the grey dots the users can inform the system about their actual position and resolve the ambiguity of location, thus allowing the system to switch back to the topological presentation of figure 2B.

In the worst case there is only very rough or no information about the actual position and orientation and the system cannot align the map to the user's actual walking direction anymore. Now a greater portion of the map has to be selected, which may include several (especially already passed) turns of the user (see figure 2D). Instead of including small landmarks that are only relevant at a single decision point, global landmarks, such as stairs or elevators have to be represented in the presentation. Since it is important to explain to the users that they cannot rely on the orientation of the map, the presentation contains a North Symbol to underline the external frame of reference. Again the user can communicate the current position to the system by clicking on the grey dots, resulting in a closeup of that area of the building. But in order to align the map to the walking direction, the system has to ensure the user's correct orientation. This task can be accomplished by advising the user to reorient towards a landmark, e.g. by prompting a text: "Turn around until the stairs are to your left and the lift is to your right" . In the outdoor scenario we use a clip-on display attached to regular glasses, so we can only present sketch like graphics on the strength of the display's size. Nonetheless, we could adapt the presentation depending on the quality of the GPS data. On the one hand we can vary the amount of detail displayed in the map. By zooming in we can switch from a survey perspective to a detailed presentation. On the other hand we can display different

[2]although they might allow for adaptation to less than the desired information quality

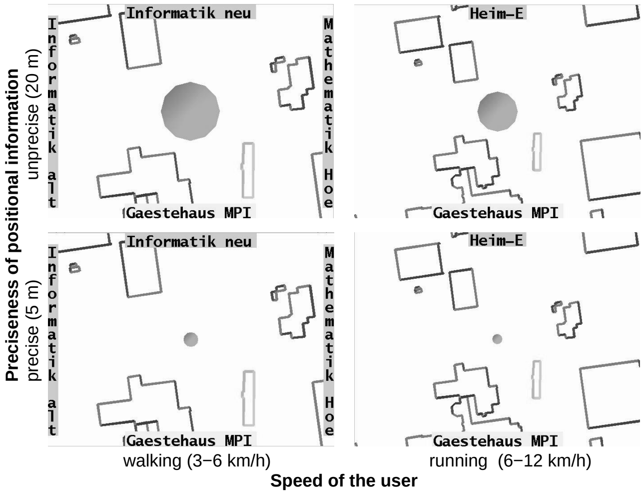

Figure 3: Four different graphical presentations in the case of active location sensitivity

textual and graphical annotations, e.g. the names of streets or buildings. Navigational instructions will be presented by means of arrows or we could visualize the appropriate part of the route from a bird's eye view. In the bird's eye view we further code the quality of positional information into the visualization of the users' current position by the size of the dot. If the quality of position information decreases the size of the dot increases (compare the upper vs. the lower part of figure 3). In addition we use the measured speed of the user to adapt the information provided to the users. If they move fast (e.g. they are running) we display a larger part of the map in order to provide a better overview and we reduce the textual annotations down to the essential information the user needs (compare the left vs. the right part of figure 3).

5. CONCLUSION

Adapting location-related services to a constantly changing and complex environment is a tough problem. In this paper, we tried to envision what level of adaptation is achievable. We were able to identify some fundamental requirements for the provision of adaptive services in mobile assistants. Based on a short review of the current state of the art, we propose an adaptive framework for location-aware mobile computing that can be used to realize the envisioned services. It allows for the integration of various approaches (such as ubiquitous computing and client-server architecture) and different technologies (such as active and passive localization). Parts of the framework are already implemented, and we plan to incorporate more and more of the features described above and to eventually realize a mobile assistant that fulfills most of the requirements we mentioned.

6. REFERENCES

[1] E. André, J. Müller, and T. Rist. Wip/ppp: Knowledge-based methods for fully automated multimedia authoring. In *EUROMEDIA'96*, pages 95–102, London, UK, 1996.

[2] C. Perkins, E. Royer, and C. Das. MANET Group Internet Draft: Ad Hoc On Demand Distance Vector (AODV) Routing, March 2001.

[3] A. G. Cohn. Calculi for qualitative spatial reasoning. In J. Calmet, J. A. Campbell, and J. Pfalzgraf, editors, *Artifical Intelligence and Symbolic Mathematical Computation (LNCS 1138)*, pages 124–143. Springer, Berlin, 1996.

[4] Y. Ding, C. Kray, R. Malaka, and M. Schillo. RAJA-A Resource-Adaptive Java Agent Infrastructure. In *Proc. of the 5th International Conference on Autonomous Agents (Agents 2001)*, 2001. To appear.

[5] Jörg Eberspaecher and Hans-Jörg Vögel. *GSM Global System for Mobile Communication.* Teubner, Stuttgart, 1997.

[6] *Foundation for Intelligent Physical Agents. Specifications*, 1998. http://www.fipa.org.

[7] ITU. GSM recommendation: 06.10, Version 3.1.2, Sep 1988.

[8] C. Kray. The benefits of multi-agent systems in spatial reasoning. In *Proceedings of FLAIRS 2001*, page to appear, 2001.

[9] A. Krikelis. Location-dependent multimedia computing. *IEEE Concurrency*, pages 13–14, April-June 1998.

[10] Alexander Leonhardi and Kurt Rothermel. A comparison of protocols for updating location information. Technical Report TR-2000-05, Universität Stuttgart, Fakultät Informatik, Germany, March 2000.

[11] C. Perkins. RFC2002: IP Mobility Support, October 1996.

[12] J. Tourrilhes. Linux Wireless LAN Howto, June 1998.

[13] M. Wooldridge. Intelligent agents. In G. Weiss, editor, *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*, pages 21–35. The MIT Press, 1999.