Research Issues in Developing a Platform for Spatial-Aware Applications

Peter Coschurba, Uwe Kubach, Alexander Leonhardi

University of Stuttgart
Institute of Parallel and Distributed
High-Performance Systems (IPVR)
Breitwiesenstr. 20 - 22
D-70565 Stuttgart
Germany
email: <firstname>.<lastname>@informatik.uni-stuttgart.de

1 Introduction

Applications that take into account the location or the context of their users are growing in importance. Examples for such applications are “Guide” [Davis et al. 1998], an interactive tourist guide developed at Lancaster University or Stick-e-Notes (University of Kent) [Brown 1995], a kind of virtual PostIt, that allows to annotate a context with an object (e.g. a notice). Context-aware applications are developed isolated from each other. No sharing of components or interaction between the different applications is possible. We propose an open platform for such applications. Such a platform would allow an easier implementation of such applications and allow a sharing of components. In the development of such a platform a number of issues from various research areas arise. In the interdisciplinary Nexus project we are currently developing such a platform.

The remainder of this paper is structured as follows: The basic idea of the Nexus project will be discussed in the next section. Section 3 will introduce the proposed architecture of the Nexus platform. In the following section we will take a look at specific research topics. We finally conclude the paper in section 5.

2 Basic Idea

We envision a platform that supports different kinds of spatial-aware applications. Such applications can support the users in many different situations. To demonstrate this, we describe a scenario of a business person traveling to a foreign city and visiting a fair there. After the arrival the Nexus system navigates its user from the airport to the fairground. There he/she can buy a ticket at a virtual ticket booth without waiting in line in front of the real ticket booth. Within the area the user can get information about the exhibitors through situated information applications or they can leave messages for the exhibitors by attaching virtual PostIt-notes to exhibits. For the lunch break the user can ask the system for nearby restaurants. On the way to the restaurant he/she can get information about historic buildings by just pointing at them with a special hardware device, a so called “telefinger”.

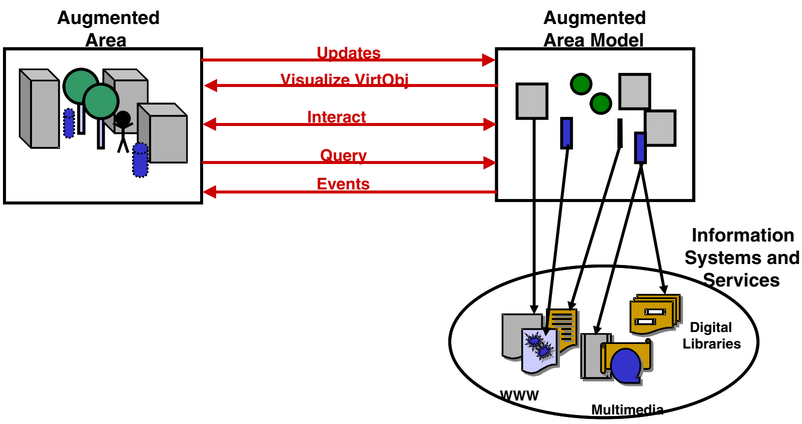

Figure 1 Basic concept of the Nexus platform.

Our basic concept (see Figure 1) to realize this vision is the Augmented Area Model (AA-Model). AA-Models are spatial models of the physical world, which can be augmented by virtual objects. Virtual objects are objects which do not represent real-world objects, i.e. there is no corresponding equivalent in the real world. The Nexus system assures that changes in the real world are automatically propagated to the models. The models enable the user to easily interact with real-world objects through the objects’ representatives in the model. For example, the users can annotate every object that is represented in a model. The Nexus system also supports queries to the models, e.g. where an object is located. Furthermore, the system has an event service that can notify the users about special events, e.g. a mobile object meeting another. A further advantage of this concept is that existing information systems and services can be easily integrated into the Nexus system by adding links from an object's representative to the appropriate information or service.

The objective we pursue is to achieve a global coverage with the AA-Models. However, initially we will only have a few AA-Models, each of them covering a spatially limited part of the physical world. The AA-Models may overlap or include each other, e.g. when modeling different levels of details. The Nexus platform offers a framework for these separate models that integrates them into one global Augmented World Model (AW-Model). In this way, a potentially global coverage can be achieved incrementally, with more and more AA-Models being integrated into the AW-Model.

3 Architecture of the Nexus Platform

The proposed architecture for the Nexus platform with its main components is shown in Figure 2.

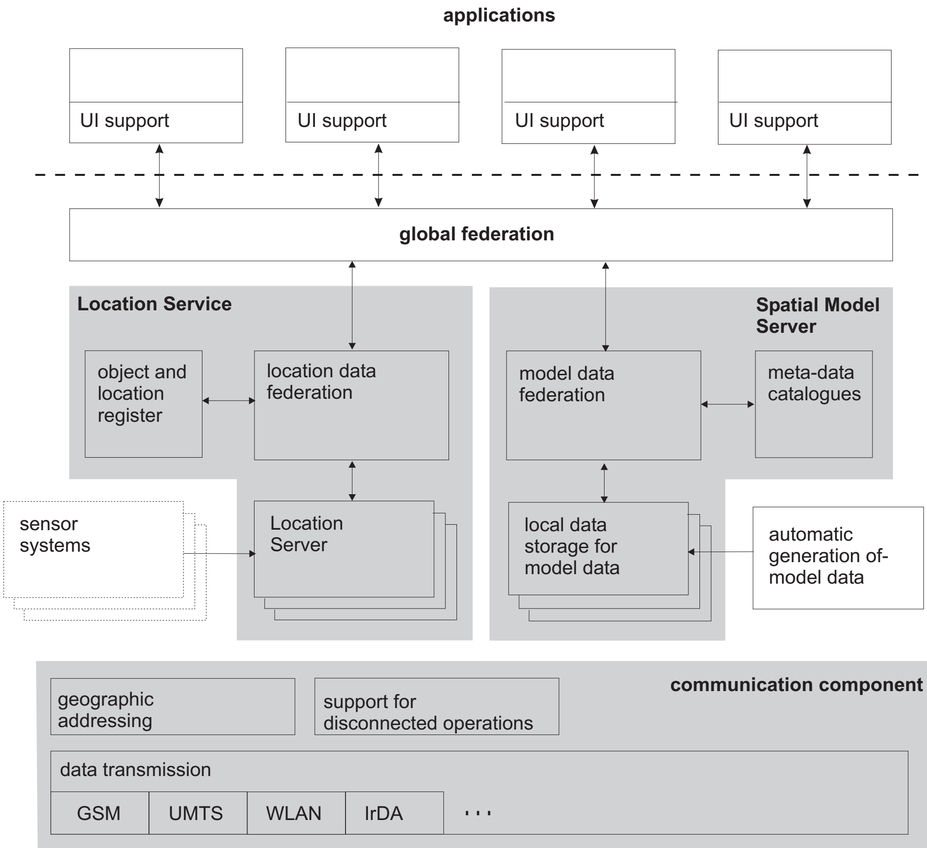

Figure 2 Main components of the Nexus platform.

The management of the information of the AW-Model is split into a *location service*, which stores the highly dynamic location information for the mobile objects, and *spatial model servers,* which store the more static data of the spatial model for objects in the real world (e.g. its buildings), the data for modeling virtual objects and links to the attached items of information. Challenges are the development of generic data-structures for describing real-world and virtual objects, where the data is stored in different levels of detail to allow efficient querying and presentation according to the requirements of the applications [van Oosterom 95]. Important is also the efficient management of the dynamic location information and of the spatial information. Because the spatial data can be rather voluminous, depending on its level of detail, methods to *generate* and to update it as automatically as possible has to be developed.

As the access to the AW-Model requires a high degree of interaction with the server it is stored on, different parts of the model will have to be managed by local location or spatial model servers. A *federation* of the location and of the spatial data allows a transparent access to the data, using *registers* respectively *catalogues* with meta-data, to find the appropriate server(s). A *global federation* component provides a global Application Programming Interface (API) for accessing the Nexus platform. Queries to this API are distributed to the location service and spatial model servers and their results are integrated accordingly. *Applications* can access the Nexus platform through this API using special libraries.

Changes in the real world are acquired through different types of *sensor-systems*, especially to determine the location information for mobile objects (e.g. GPS-sensors or Active Badges). As there is no ideal location sensor, which e.g. works in-doors as well as out-doors, it is necessary to combine the output from different types of sensors and propagate it to the model. A research challenge here is to use different types of sensors simultaneously (e.g., a GPS together with an inertial sensor) to improve the availability and accuracy of the location information. Especially for the positioning in-doors, it is also planned to integrate image analysis.

The communication between different components of the platform and with its applications is handled by a specific *communication platform*. This component integrates different wireless communication mechanisms and provides a single interface to them. It also allows a seamless transition from one medium to another. Moreover, the communication component allows to request a certain quality-of-service and initiates renegotiations if it cannot be provided anymore. *Geographic addressing* allows, besides the conventional addressing schemes uni-, multi- and broadcast, to send a message to the mobile objects inside a given area. This mechanism is very useful for applications as well as for location-aware system mechanisms. Furthermore, a hoarding mechanism, in which information, which is most likely to be used in the future, is transmitted to a mobile device in advance, will be used to overcome temporary *disconnections* of a wireless network.

A global issue is to guarantee the security and the privacy of personal information stored in different components of the platform. As some of this information, especially the location information of its users, is very sensitive, the platform has to enable its users to limit access to this information and contain mechanisms to guarantee the security of the data.

4 Special Research Issues

The Nexus project is concerned with different issues of Mobile Computing, namely the efficient addressing of all mobile objects inside a given area, improving the access to location-aware information over a wireless network through hoarding mechanisms and the managing of the location information for mobile objects.

4.1 Geographic Addressing

The use of location information offers new possibilities regarding message distribution. With location-aware applications it should be possible to send a message to all recipients in a specific geographic area. The only work on this subject conducted so far is done by Navas and Imielinski (see [Navas & Imielinski 1999] or [Navas & Imielinski 1996]) as part of the Dataman project. They use a two-dimensional addressing scheme that relies solely on geographic coordinates. In [Navas & Imielinski 1998] they describe a prototypical implementation of their geographic routing approach. Features like tree-dimensional addressing or scalability in large environments have been not achieved yet. So additional research has to be conducted in order to have a satisfying solution for geographic addressing and message forwarding for a platform like Nexus.

The issues we are looking into right now are:

- Addressing: In a system like Nexus, where a model of the real world is available, it makes sense to use the information in that model for addressing. So one should be able to specify a room as destination for the message and all the recipients in that room would get the message. There is no need for the user to enter or even know the exact geographic coordinates of that room.
- Scalability: An important issue is the scalability of the geographic addressing and message propagation. The solution has to scale well, so that it can not only be used in a small setting, but also in large maybe even global scale implementations.
- Semantics: All methods for geographic message propagation developed provide a "best effort" message delivery. No work has been done so far on what semantics are required for geographic message propagation, and how this can be achieved. We plan to define different semantics and find means to provide them.

4.2 Context-Aware Hoarding

When information is accessed through a wireless WAN, the users often have problems with low bandwidth, high delays and frequent disconnections. Hoarding is an efficient method to overcome these problems. The idea of hoarding is to transfer the information a user will probably need in the near future to the mobile device before it is actually accessed.

The Nexus platform will be based on an info-station infrastructure as proposed in [Badrinath et al. 1996]. There, a number of high-bandwidth wireless local LANs with a low range, so called info-stations are placed in an area otherwise only covered by a wireless WAN (see Figure 3).

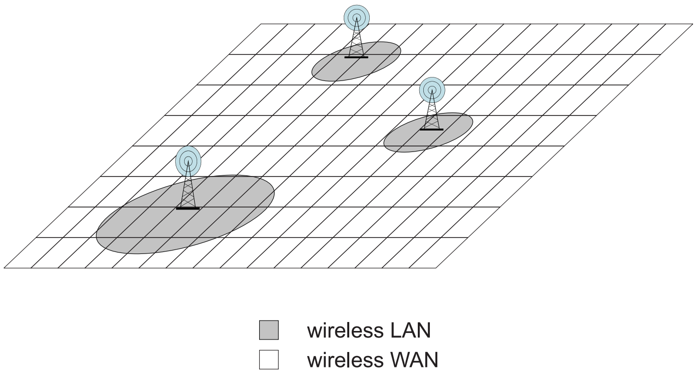

Figure 3 Info-station infrastructure.

Our hoarding mechanism uses this infrastructure as follows: When a user arrives at an info-station it tries to hoard as much as possible of the information he/she will probably need before reaching the next info-station while meeting the restrictions in transfer time and storage space available on the mobile device. If the user is not in an area covered by an info-station, no hoarding is performed at all. So the users only suffer from the disadvantages of wireless WAN connections, when they need information that was not hoarded at an info-station. Ideally, they do not need the wireless WAN at all.

The problem with hoarding is to predict which information items a user will need in the future. In spatial-aware applications it depends strongly on the users' location, which information items they request. Therefore, we use information about a user's future location in order to determine the information items to hoard. A second criterion we use is the average number of information items the users request while staying at a certain location.

Our mechanism gets the information for its hoarding decision from two different kinds of sources. The first kind of sources are external sources, i.e. the application through which the users access the Nexus platform and the users themselves. For example a navigation application can provide exact predictions on a user's future location. In [Kubach & Rothermel 2000] we introduced hoarding patterns as means to specify knowledge about the locations a user will probably visit. The second source is information gathered by the mechanism itself. To gather this information the mechanism uses a monitor component that observes the users' behavior, i.e. the probabilities with which the users visit certain locations and the number of information items they request. To assure the users' privacy all information is stored anonymously.

A theoretical analysis of our mechanism's performance showed that considering the users' context when deciding which information items to hoard is highly beneficial. However, we currently lack an evaluation in a real scenario. Such an evaluation will require a field trial with a considerably large set of real users. Before we start such a field trial, our next step will be the development of a computer model that allows us to simulate the users' movement patterns and information access patterns in different scenarios.

4.3 Managing Location Information

A very important aspect of the Augmented World Model is the knowledge about the exact location of the mobile objects, especially of its users. As this information is often highly dynamic, depending on the movement characteristics of the object, we require special mechanisms and a special location service for handling the location information. This service will have to handle a high number of updates, so special update protocols have to be used to reduce them as much as possible.

In [Leonhardi & Kubach 1999], we have proposed an architecture for such a location service. It will basically offer position queries (i.e., querying the current location of a mobile object), range queries (i.e., finding all objects inside a given area) and an event mechanism, which sends a notification if a certain situation occurs (e.g., if a user enters an area). The location information for the mobile objects is gathered from different types of sensor systems. This can for example be a GPS sensor attached to a personal device a user is equipped with. It can also be an Active Badge system (see [Want et al. 1992]), with badges attached to pieces of office equipment.

The location information for all mobile objects inside a given area are stored on a certain location server, which is associated with this area. All sensor systems that acquire location information for this area report it to this location server. An application sends its queries to the nearest server, which answers it if the information is available locally. Otherwise, it forwards the query to the appropriate server(s), which it determines through the object and the location register. The object register stores the attributes of a mobile object, like its maximum or average speed, and knows on which location servers its location information is currently stored. The location register stores the information about which location server is responsible for which area.

Our future work in this area will focus on the following points:

- Different configurations of the architecture for the location service will be compared depending on different scenarios. The scenarios will differ in the number of mobile objects, as well as in their mobility and querying characteristics.
- In [Leonhardi & Rothermel 2000] basic update protocols for transmitting location information efficiently and effectively have been compared. We hope to be able to further improve the efficiency by using different types of dead-reckoning protocols, where the server estimates the current location of the mobile object by some means. The client calculates a similar estimation and sends an update when the actual position differs from the estimated one by more than a certain degree. In contrast to [Wolfson et al. 1999] the mechanism we plan to develop will not assume prior knowledge about the route of the mobile object.
- Event notification will be handled by a distributed event management component, which is responsible for detecting the occurrence of global predicates. These are transformed into basic predicates, which are then assigned to certain location servers. Based on the notification for these basic predicates, the event management decides if to send a notification for the global predicate.
- An important aspect of the location service is the security and the privacy of the location information. Here, a user shall be able to control in detail, to whom and to what degree his/her location information can be given.

5 Conclusion

We believe that there is a need for a common and open infrastructure for spatial-aware applications as the demand for spatial-aware applications will grow continuously. A platform as proposed would allow an efficient implementation of such applications. Moreover, it would make the information gathered by the sensors available to all the applications and would enable them to cooperate and exchange data. The ability to switch the level of detail of the model and the ability to interact with the real world, without having special hardware, are other benefits of the proposed platform.

We have presented some of the open research issues that have to be solved in order to be able to provide such a common platform. We believe that, once a common platform for spatial-aware applications is available, it might become as important and widespread for spatial information as the WWW is today for conventional electronic information.

6 References

[Pascoe 1997] Pascoe, J.: The Stick-e Note Architecture: Extending the Inteface Beyond the User, Proceedings of 1997 Int. Conf. on Intelligent User Interfaces, 1997

[Davis et al. 1998] Davies, N.; Mitchell, K.; Cheverst, K. and Blair, G.: Developing a Context Sensitive Tourist Guide, Proceedings of the First Workshop on Human Computer Interaction for Mobile Devices, Department of Computing Science, University of Glasgow, U.K., pp64-68 21-23 May 1998

[Badrinath et al. 1996] Badrinath, B.R.; Imielinski, T.; Frenkiel, R., and Goodman, D.: Nimble: Many-time, many-where communication support for information systems in highly mobile and wireless environments, http://www.cs.rutgers.edu/~badri/dataman/nimble/.

[Kubach & Rothermel 2000] Kubach, U.; Rothermel, K.: An Adaptive, Location-Aware Hoarding Mechanism, Proceedings of the Fifth IEEE Symposium on Computers and Communications (ISCC 2000), Antibes - Juan les Pins, France, to appear.

[Leonhardi & Rothermel 2000] Leonhardi, A.; Rothermel, K.: A Comparison of Protocols for Updating Location Information, Technical Report TR-2000-05, Department of Informatics, University of Stuttgart, 1999.

[Leonhardi & Kubach 1999] Leonhardi, A.; Kubach, U.: An Architecture for a Universal, Distributed Location Service Proceedings of the European Wireless '99 Conference, Munich, Germany, pp. 351-355, ITG Fachbericht, VDE Verlag, 1999.

[Navas & Imielinski 1999] Imielinski, T.; Navas, J. C.: GPS-Based Geographic Addressing, Routing and Resource Discovery, Comunications of the ACM Vol. 42, April 1999.

[Navas & Imielinski 1996] Imielinski, T.; Navas, J. C.: GPS-Based Addressing and Routing, IETF RFC 2009, November 1996.

[Navas & Imielinski 1998] Imielinski, T.; Navas, J. C.: Multi-hop Dynamic Geographic Routing, Technical Report (LCSR-TR-364), Rutgers University, May 1998.

[Wolfson et al. 1999] Wolfson, O.; Sistla, A., P.; Chamberlain, S.; Yesha, Y.: Updating and Querying Databases that Track Mobile Units. Distributed and Parallel Database Journal, 7(3), pp. 1-31, 1999.

[van Oosterom 95] van Oosterom, P.: The GAP-tree, an approach to 'on-the-fly' map generalization of an area partitioning. in: J.-C. Müller, J.-P. Lagrange & R. Weibel, Hrsg., *GIS and Generalization - Methodology and Practice*, Taylor & Francis, pp. 120-132, 1995.

[Want et al. 1992] Want, R.; Hopper, A.; Falcao, V.; Gibbons, J.: The Active Badge Location System. ACM Transactions on Information System, 10(1), pp. 91-102, 1992.