

BerlinTainment: An Agent-Based Serviceware Framework for Context-Aware Services

Jens Wohltorf, Richard Cissée, Andreas Rieger, Heiko Scheunemann

DAI-Labor, Technische Universität Berlin, Salzufer 12, 10587 Berlin, Germany

{jens.wohltorf, richard.cisse, andreas.rieger, heiko.scheunemann}@dai-labor.de

Abstract

In the near future, providers of mobile services will face increasing competition. Therefore, the ability to design, develop and deploy reliable context-aware services fast and easily will become essential. We introduce an agent-based Serviceware Framework assisting service providers in developing innovative services, thus reducing the time-to-market of the respective applications. The framework offers personalization, location awareness, and device independence for each single service. This paper describes the utilization of the different modules of the framework as well as prototypical application services in the entertainment domain based on the framework developed within the BerlinTainment project.

One of the realized services is exemplarily described in this paper. The Intelligent Day Planner allows the user to plan his entertainment activities in an intelligent and efficient manner. Based on the users' individual preferences, an individual day plan is recommended. Based on this day plan, additional service may be utilized including reservation, time table organisation, and route finding.

Keywords: Serviceware Framework, multi-agent systems, mobile service, service engineering, device independence, personalization, context awareness

I. Introduction and Overview

Mobile services can be defined as the context-aware supply of numerous content to the end-user, who utilizes different devices and network

technologies independently. They have to cover countless requirements in order to successfully attract end-users and achieve their retention. Thus, mobile services have to operate in a time-, location- and situation-based manner anywhere, anyhow, and any time, thereby creating an added value for the end-user. Moreover, mobile services have to be developed in an effective and efficient way to achieve amortisation of the enormous investments. For such a development, a powerful service engineering methodology has to be applied [Albayrak, Wohltorf, et al 2003].

Context-aware services include solutions for three main issues in mobile computing: ubiquitous access, provision of personalized information, and location-based services. Ubiquitous access in this context refers to the concept of “anytime, anywhere” access to services. Personalized and location-based services present relevant information only, based on the user’s preferences and current position, thus enabling the user to counter the threat of information overload. Because of the highly distributed and dynamic character of context-aware services, utilizing agents for the tasks of managing, providing, and updating these services is a highly suitable approach.

The paper is structured as follows: the following section outlines the major motivations underlying research activity in this area; it is followed by a survey of the relevant state of the art. The core of our work, the BerlinTainment Serviceware Framework, is presented in Section IV. In Section V, details of our specific prototype for the entertainment domain are described. In the final section, conclusions are drawn from our experiences.

II Motivation

The objective of the BerlinTainment project is to provide information over the Internet in an easy and usable manner. For this purpose, a scalable Serviceware Framework has been developed, which directly connects end users with service providers. The framework is based on Multi-Agent System technology and utilized for the development of context-aware services, i.e. services providing personalized, location-based information on different end-user devices. As a showcase for the functionality of the Serviceware Framework, different prototypical services in the entertainment domain have been developed and integrated into the BerlinTainment demonstrator.

These services provide various benefits to Berlin's visitors, new residents, tourists or business travellers, keeping them informed about interesting leisure activities and background information. All information is context-based, i.e. based on the current user's location (including the current time), situation, and profile. The BerlinTainment services may be used to plan comprehensive day itineraries, find adequate restaurants, shows, music events, and movies, make reservations, rate locations for other users, view, adapt or delete individual profiles, and determine current locations, points of interest, and routes.

III State of the Art

Several projects providing services to end-users have been developed that are similar to BerlinTainment. One of them is the German DOM project, which aims to provide personalized and location-based services for end-users [DOM 2002]. The approach of DOM consists of four layers: Content, Basic Functions, Transactions and General Service layer. The DOM approach is further structured in functional components but lacks in contrast to agent-based approaches in terms of scalability and flexibility.

The same disadvantage applies to the LoVEUS project [Loveus 2003]. LoVEUS aims to provide ubiquitous services for personalised and tourism-oriented multimedia information. It follows a client-server approach incorporating a server with numerous clients.

The CRUMPET approach deals with the creation of user-friendly, personalized, and mobile tourism services [Crumpet 2003]. The architecture is based on a multi-agent system. The user or terminal agents are hosted on the end-users'

terminal devices and provide service GUIs. A brokerage function enables the user agents to declare interest in particular services and receive information about services that meet special criteria, such as proximity constraints. For example, when the users' location changes, local services may then meet the specified service constraints and are offered to the user.

While personal agents are generally suitable for context-aware services, hosting agents on users' devices has the drawback that special software is required on the device. Thus the range of usable devices is limited and device independence cannot be achieved.

IV. Serviceware Framework for Context-Aware Services

The BerlinTainment project focuses on the realization of a scalable Serviceware Framework based on Multi-Agent System (MAS) technology. For an introduction to MAS technology, we refer to [Albayrak 1998]. Basically, MAS architectures consist of agents encapsulating specific functionality and offering services to exchange information with other agents. All agents exist within specific environments, the agent platforms. The interaction of agents is based on ontologies defining a common vocabulary. The Serviceware Framework is utilized for the development of context-aware services, i.e. services providing personalized and location-based information on different end-user devices. As a showcase for the functionality of the Serviceware Framework, different prototypical services in the entertainment domain have been developed and integrated into the BerlinTainment demonstrator.

The Serviceware Framework is based on a Multi-Agent System architecture for the following reasons: MAS architectures fulfil the requirements on frameworks for context-aware services regarding

- **Modularity:** For different context-aware services and scenarios, it should be possible to use only those parts of the framework that are actually required. MAS-based applications are mainly configured by selecting and defining the participating agents. Therefore different modules made up by groups of agents may be changed easily.
- **Scalability:** The framework should be usable for small, non-public systems as well as for applications with a large target audience. In MAS architectures, scalability is mainly

achieved by duplicating the agents responsible for critical tasks, thus distributing the load between multiple identical agents and removing bottlenecks.

- **Adaptability:** The functionality provided by the framework should not be static, or it would be outdated soon. Therefore it should be possible to add, remove, or replace parts of the framework, if possible even within deployed systems, without requiring changes on the framework user's side. MAS-based applications may be reconfigured at runtime, i.e. agents may be added or removed to adapt the functionality provided. The newly offered services may be used immediately.
- **Distributedness:** It should be possible to distribute the framework e.g. to increase the overall robustness and security. Mobile agents have the ability to migrate between platforms which may be located on different servers.

The framework consists of several sub-modules, each offering agent services to applications utilizing the framework, as shown in Figure 1:

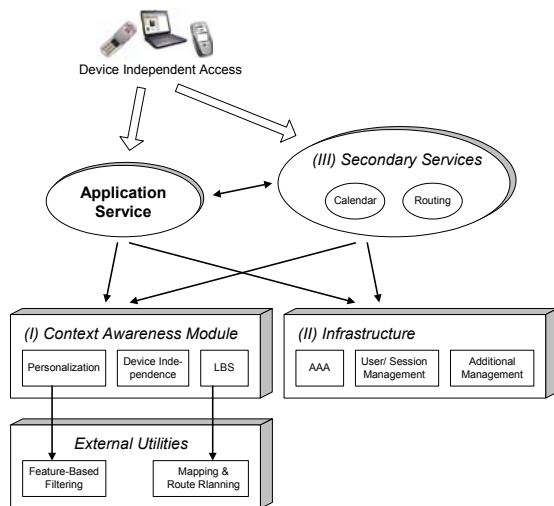


Figure 1: Architecture of the framework for context-aware services. The black arrows indicate usage of agent services, the white arrows indicate User Interaction.

The **Context Awareness** module represents the nucleus of the framework. Its sub-modules provide services within the areas of

- **Personalization:** Provides access to Information Filtering techniques processing large amounts of information, recommending to users only individually relevant information, based on user profiles.

- **Device-Independence:** Provides services utilized to generate User Interfaces for different devices without having to change the underlying functionality of the respective services.
- **Location-Based Services (LBS):** Provides services supporting the localization of users, and the processing of location information, e.g. in order to determine distances between locations or transform location co-ordinates.

External Utilities modules provide functionality accessible via the context-aware services modules, utilizing interfaces to existing applications, such as Information Filtering techniques (feature-based filtering, knowledge-based filtering) or location and routing software.

The **Infrastructure** module supports, among other aspects, the management of users, sessions, and services. Sensitive information, such as personal user information, is protected from unauthorized access by encapsulating it within dedicated user agents under control of the respective users.

Secondary Services modules provide additional services utilizing parts of the context-aware services modules, such as calendar, scheduling and notification services, and services allowing the configuration of user profiles. A special secondary service provides a single access point to application services.

The Implementation of the framework has been carried out based on the FIPA-compliant MAS-architecture *Java Intelligent Agent Componentware* (JIAC) [Sessler, Albayrak 2001] [Fricke, et al 2001]. JIAC integrates fundamental aspects of autonomous agents regarding proactiveness, intelligence, communication capabilities and mobility by providing a scalable component-based architecture. Additionally, JIAC offers components realizing management and security functionality, and provides a methodology for *Agent-Oriented Software Engineering* (AOSE).

Certain aspects of the framework are addressed by JIAC itself: Regarding e.g. device-independent access to services, JIAC provides *Multi-Access Agents* (MAA) capable of adapting user interfaces, based on the user's current device. Basically, the CC/PP profile of a device is used to determine the language the abstract user interfaces are to be transformed into (such as HTML/ WML for browser-based interfaces, or VXML for voice-based interfaces), and to determine the way

graphics are to be presented, based on the device's display size.

V. Prototypical Application Services

The Serviceware Framework is currently utilized by prototypical application services providing information about restaurants, movies, concerts and other types of entertainment. Based on a users' prior interactions with an application service, a user profile is maintained and utilized to provide personalized recommendations and location-based information. The application services may be accessed from different devices, such as mobile phones, PDAs, or PCs via XML-based GUIs.

A special secondary service, the Intelligent Day Planner, has been implemented as a showcase for the available functionality, integrating different services. The Intelligent Day Planner allows a user to schedule different activities, such as restaurant, cinema, and theatre visits, in order to receive context-aware recommendations. The given recommendations are based on the users' preferences. The locations of the respective venues as well as the starting times of the activities are also taken into account to minimize the temporal and spatial distances between the recommended activities.

In the following, a complex BerlinTainment scenario is described, introducing a variety of realized services. BerlinTainment is accessible via <http://www.berlнтаainment.de>. If you have already registered as a user, the BerlinTainment services can be accessed via the login button. Otherwise, the registration button brings up a dialog that requires you to enter a username and a password. If you are interested in using the notification functionality of BerlinTainment you should also enter your e-Mail address for e-Mail notification, your mobile phone number for SMS (Short Message Service)-based notification, and a phone number for voice notification.

Furthermore, you have the opportunity to select potentially interesting information services. If you do so, the next dialog will ask you to initialize your individual user profile for the selected categories, as shown in Figure 2.

For using the full potential of personalized BerlinTainment services, your profile should be set up at this point. In order to obtain personalized recommendations, you may edit your preferences for the selected categories in this dialog. Additionally, you may specify the importance of

Figure 2: Profile Initialisation

each attribute. If you leave out some category values these categories will not be considered.

After entering your preferences, you may access the BerlinTainment Service Zone. Here, you may choose, among other options, the "Intelligent Day Planner".

Figure 3: Intelligent Day Planning

The Intelligent Day Planner is used to plan daily activities in an intuitive way. You do not have to search all entertainment activities separately. Instead, you may choose a day and time for a combination of different activities, such as visiting a restaurant, cinema, or theater, as shown in Figure 3. Based on your user preferences (i.e. the personal profile) and your desired schedule, personalized recommendations are generated. These recommendations are presented in the result dialog, as shown in Figure 4, allowing you to optimize your schedule in different ways: You may have a look at the details of any given recommendation or change recommendations by searching for better alternatives.

Your recommended schedule for Thursday May 6, 2004, based on your appointments and planned activities:

17:00

→ SHOW DETAILS
→ GET NEW RECOMMENDATION
→ SEARCH FOR ALTERNATIVE
→ REMOVE THIS ACTIVITY

Eat out at the restaurant "Reinhard's im Nikolaiviertel". The Address is: Poststrasse 28 in Mitte. This restaurant offers International cuisine. Phone: +49 (0) 30 / 2425295.

↕ SHOW ROUTE BETWEEN THESE LOCATIONS

19:00

→ SHOW DETAILS
→ GET NEW RECOMMENDATION
→ SEARCH FOR ALTERNATIVE
→ REMOVE THIS ACTIVITY

See the movie "Peter Pan" starring Jason Isaacs, Jeremy Sumpter, and Rachel Hurd-Wood at the cinema "Cubix UFA-Palast Berlin Alexanderplatz". The Address is: Rathausstr. 1.

↕ SHOW ROUTE BETWEEN THESE LOCATIONS

21:30

→ SHOW DETAILS
→ GET NEW RECOMMENDATION
→ SEARCH FOR ALTERNATIVE
→ REMOVE THIS ACTIVITY

See the show "Wenn der Tierse 2x klingelt" starring Dagmar Jaeger, Stefan Martin Müller, and Michael Nitzel at the venue "Distel". The Address is: Friedrichstr. 101.

→ SHOW MAP FOR ALL → MAKE RESERVATIONS
→ RETURN TO SERVICE ZONE

Figure 4: Day Planner Recommendations

Back in the day planner result dialog, you may access a map showing the locations of the different activities and the route between these locations.

The map dialog allows you to navigate in the map and provides a textual description of the route between the given locations. If you finally accept the given recommendations, you may add them as permanent appointments to your calendar. This concludes our scenario.

You may decide to interrupt the interaction with the system at any point. In this case, you may

continue the session later, even on a different device, without having to restart the respective service. This is especially useful for mobile service usage: As an example, a service may be started via a PC. After the route to a recommended location has been determined, the service is continued from a mobile device on the way to this location to ascertain the route.

VI. Conclusion

The requirements of developing services based on high-level demands and concepts have steadily risen in the last few years. This accompanies the rising costs, longer development times, and the increasing expense of constructing next-generation services for satisfying the growing demand of today's customers. Service components are now distributed on different computers and must also be accessible over different networks. This presents new challenges to both the service developer and the network operator.

In order to meet these challenges, new concepts and technologies have been implemented that, as opposed to classical approaches, are specifically intended for implementing complex services. Here, key factors are intelligent support for flexible systems as well as interoperability between products in order to give the user straightforward access to services.

Intelligent software agents are a basis technology for the construction of flexible context-aware applications. In addition to the concepts of object-oriented programming the software agents also incorporate aspects of artificial intelligence and communications systems, providing a wide-ranging approach satisfying complex requirements.

The agent architecture JIAC IV has been used for the implementation of BerlinTainment's Serviceware Framework, because it allows the service developer to create distributed, adaptable, and modularized services that satisfy the requirements of the service provider, network operator, and end-user. A prototypical demonstration application based on the generic functions of the system has been successfully implemented and evaluated. We have shown that multi-agent system technology is suitable for developing context-aware services including personalization (i.e. profile-based information searching and filtering), location-relevance of presented information, and device independent access to mobile services.

References

[Albayrak, Wohltorf, et al 2003] Albayrak, S., Wohltorf, J., Varone, N., Rieger, A. (2003), "Service Engineering - Methodology for 3G beyond 3G Service Development", working paper.

[Crumpet 2003] CRUMPET - Creation of User-friendly Mobile services Personalised for Tourism (2003), <http://www.ist-crumpet.org/>

[DOM 2002] DOM – Der Orientierte Mensch, project funded by German Federal Ministry of Education and Research (2002), <http://www.der-orientierte-mensch.de/>

[Loveus 2003] LoVEUS - Location aware Visually Enhanced Ubiquitous Services (2003), <http://loveus.intranet.gr/>

[Albayrak 1998] Albayrak, S. (1998), Introduction to Agent Oriented Technology for Telecommunications. In: S. Albayrak (ed.): Intelligent Agents for Telecommunications Applications, IOS Press, p. 1 – 18, 1998.

[Sessler, Albayrak 2001] Sessler, R., Albayrak, S. (2001), Serviceware Framework for Developing 3G Mobile Services. The Sixteenth International Symposium on Computer and Information Sciences, ISCIS XVI

[Fricke, et al 2001] Fricke, S., Bsufka, K., Keiser, J., Schmidt, T., Sessler, R., Albayrak, S. (2001), Agent-based Telematic Services and Telecom Applications. Communications of the ACM, April 2001.