Integration of Mobile Computing in Applications for the Service Sector – Design and Implementation

Klaus-Georg Deck and Herbert Neuendorf

University of Cooperative Education, Berufsakademie Mosbach, Lohrtalweg 10, D-74821 Mosbach, Germany

Abstract. For representative applications from the service sector, scenarios for mobile systems are introduced, the similar structure of which can be described with a common pattern. A prototype of its implementation is introduced based on Web Services in Java, the landscape of the most important classes is described. Technical implementation is done with a Blackberry handheld from T-Mobile/RIM using push technology within the GPRS network.

1 Introduction

Due to the rapid development of telecommunication technology, widespread use of mobile terminals and the increasing acceptance and use on the user side, new scenarios for mobile application systems are arising in all sorts of areas.

Especially in the service sector, frequently services are rendered by employees acting on-site so that underlying business processes can be universally mapped using a widely available IT infrastructure.

Existing applications based on a client server or Internet architecture are enhanced with mobile subsystems or they are being replaced with conventionally implemented components.

While the technological basis for supporting value chains beyond corporate boundaries has been created up to now using the Internet, nowadays Internet-based standards, and especially Web services, are providing an infrastructure for integrating mobile components.

More and more powerful mobile terminals create the technological prerequisites for the use of these standards and in the meantime, they have become a suitable alternative to notebooks in regard to interface design and user interaction.

In the following, representative service scenarios with employees that are working on-site will be introduced and examined in regard to their potential for mobile business process optimization. Then the structures that underlie these processes are determined. These structures can be used for a standardized solution for integration of mobile components.

Deck K. and Neuendorf H. (2005).
Integration of Mobile Computing in Applications for the Service Sector – Design and Implementation.
In *Proceedings of the 4th International Workshop on Wireless Information Systems*, pages 71-80
DOI: 10.5220/0002541500710080

2 Application Scenarios

2.1 Rescue Services

Initial Situation

After an emergency call comes into the rescue control center, the relevant data (location of accident, accident situation, estimated number of injured, etc.) is gathered in a control system and is sent directly to an available EMT team by telephone using the analog radio network.

The rescue team (emergency physician, EMT) proceeds to the site of the accident and begins emergency care for the injured and transports them to the appropriate hospital – usually after speaking with the control center. Next, the services rendered are summarized in a service report.

Problems and Challenges

Due to bad voice quality and interfering environmental noises, incorrect information is frequently transmitted, which leads to further time-consuming queries or even to bad directions when driving to the accident location or the hospital [7, p. 156-158]. When transferring orally transmitted data in handwritten notes, there is also a danger of losing information or recording it incorrectly.

There are often problems with assignments, both on the side of the emergency team and from the control center due to frequency shortages and multiple teams having to use the same radio channels.

Generally, the radios in the ambulances are fixed to the vehicle, which means that no direct communication between the control center and the EMTs is available in accident locations that are hard to get to.

The data privacy situation is also very unsatisfactory. With voice radio, the data is transmitted without being encoded and can generally be heard by anyone. Various data protection officers are dealing with this problem (see [3]) and are pushing for a solution because emergencies usually involve highly sensitive personal information.

Mobile Solution Concept

The advised mobile solution intends to provide the EMT teams with mobile terminals that are directly connected to the centralized control system. There, in addition to the actual emergency data, the emergency vehicles and emergency team, technical and staff equipment and the current availability status are also managed.

When an emergency call comes in, the emergency data is collected in the system by the control center booker, and depending on the gravity of the accident, is assigned to one or more available emergency teams, with the relevant data being automatically transferred to the appropriate handheld. This simultaneously triggers an alarm signal.

After the team reaches the site of the accident and performs first aid on the injured parties, the availability of OPs and beds in nearby hospitals can be determined depending on the type and gravity of the injuries. Here there is no time-consuming waiting for confirmation from the hospital or control center. The prerequisite for this is that the availability information for the individual hospitals be maintained in the control system.

Result

In contrast to the conventional system, the mobile concept has the following advantages:

- Information is transmitted clearly and is constantly available on the terminal.
- The data is only sent to the responsible EMT team and third parties cannot unintentionally access it. This greatly diminishes issues with assignments.
- EMTs and control center can communicate outside the ambulance as well.
- The control center is constantly aware of the current status of the emergency team (read confirmation of the emergency notification, drive to location of accident, getting to site of accident, etc.).
- Provision of selection lists on the handheld (for example, diagnoses) enables efficient, standardized data entry.
- The control system provides support in drafting the service report because the majority of the necessary information was summarized automatically.

Mobile devices with telephone functions also enable direct communication between the emergency physicians and hospital doctors without the need for transfer by the control center.

The use of location-based services, in which GPS receivers or cell identification of the location coordinates of the EMT team can be determined and stored in the system provide more potential. This enables integration of route planning systems to determine the fastest route to the accident side or to the hospital. In addition, the distance to the accident site can be considered when the emergency team is dispatched, for example when they are on their way from the hospital to the station.

With direct addressing of the mobile terminal and the option of personal user authentication, a high degree of data protection is created – when the cryptographic process that is already established for the cellular phone network is used as well.

2.2 Outpatient Medical Services

Initial Situation

In facilities for outpatient medical services, software solutions for planning, documentation, and billing of health services have been used successfully for years. These types of systems manage the personal and medical data for patients, various catalogs of care and information on the medical qualifications and availability (time) of the caregivers. Supported by the system, various health services that have been prescribed can be summarized in a care schedule for a specific patient, and resource and service plans for individual employees can be created using it.

The services performed are documented in the care schedule – generally in paper form – and can be manually added to the accounting system later. At the end of the billing period, the insurance provider can be billed.

Problems and Challenges

The manual entry of data from the care schedules into the system afterwards requires additional personnel and the data update is delayed, sometimes by days. Illegible handwriting or abbreviations that are individual to certain employees can lead to additional questions or to incorrect information. If additional information is required for a patient on site, this has to be retrieved by telephone or personally in the control center.

At the end of the shift, in addition to the patient-specific care information the employee has to create a report that summarizes the care times for individual patients, the individual driving times and the kilometers driven in an overview. This job is often considered a burden by the employees due to the sometimes redundant content and due to the fact that it has to be done after the shift is done, that is once work is finished.

Due to the delay in the manual data update in the main system, billing of the individual insurance companies can only be done with a delay as well.

In outpatient medical services, the above problems are only enhanced by the relatively large number of low-paid employees, because coordination problems occur when several care providers are responsible for a patient with a short period of time.

Mobile Solution Concept and Result

Mobile terminals are already being used in some outpatient medical services, but data updating is still usually being done only at the control center via cradle.

In this type of scenario, employee-specific service schedules are generated by the central system with the associated patient-specific care schedules and are transmitted to the mobile devices in electronic format, where they are available while the patient is being cared for.

At the end of the care visit, the care provider documents the length of each of the services performed that are displayed on the handheld in patient-specific selection lists, where they only need to be selected. In the same place, information on how the procedures went and any other important information on the patient can be noted.

In systems with mobile synchronization, this information is sent to the control system in real time, where it is available to clerks and care providers, and as needed to other mobile employees. If additional medical information is required for the patient (medical history, further diagnoses, etc.), this can be requested from the control center directly via handheld. Further information stored in the central system, such as details for care procedures can also be sent to mobile users upon request.

The service reports for employees and care documentation for the patients is generated centrally by the system. They can be edited with word processors and then printed. The increased quality and currentness of the data leads to greatly improved quality assurance and the efficiency and effectiveness of the care procedures can be more easily checked and controlled in this way.

On-site services can also be integrated into this scenario, perhaps in the route and dispatch planning or when determining the kilometers driven.

2.3 Further Application Scenarios

Further service scenarios in which mobile computing could provide added value could include:

- Facility management
- Mobile cleaning and maintenance services
- Utilities companies
- Logistics services

2.4 Summary

The opportunities and application potential provided by mobile services that are available for companies, customers and of course, employees, can be summarized as follows:

- Currentness of information
- High quality of information
- Real-time transparency and active control of the business processes
- Business process optimization by shortening process chains and reducing integration gaps
- Faster billing – increase in liquidity and profitability
- Improved customer service – increased customer satisfaction
- Employee concentration on core competencies – increased employee satisfaction
- Security of information transmission and data processing

3 Architecture

3.1 SB2ME Pattern

All of the applications introduced illustrate clearly defined, simply structured work processes in which employees on site receive consistent, clearly structured order data that can be highly standardized, and they process it on-site and send the results of the work procedures back [11, p. 22 et seq.]. In this way, it is quite simple to show these processes in the categories of communication between a control center and mobile terminals. The origin and the starting point is the control center, where each business process can be traced back to an individual employee. Data is not exchanged between the various mobile employees, instead it is always transferred via the control center.

In the use-case scenarios for all applications, two actors can be identified: The actor *Control Center*, from which new orders are generated and the mobile employees are coordinated and managed – and the actor *Mobile Employee*, who is the unique recipient of an order and who processes it and sends it back to the control center. In addition, mobile employees ask the control center for further information, but they generally cannot change it.

The base pattern here is called the *Simple-Business-to-Mobile-Employee* Pattern (SB2ME). It is characterized by:

- Simple process and data structures (Simple**)**
- Control center of the company (Business) initiates the business process

- The main impact comes from the control center and moves toward the mobile employee (to-Mobile-Employee / 2ME)

3.2 Technical Implementation

The systems are mainly implemented with a 3-tier architecture. The data from the system is kept persistent with a relational database system for access by the control center server. The business logic runs in the Web server of the control center server with Java Servlet technology and the connected Web services. A browser-based Web interface is used as the front end for the control center and a handheld is used for the mobile employees (figure 1).

The data from orders that are not yet complete is made persistent at the client so that it is still available once the device has been switched off.
For the application introduced above, reference implementations were made on a common basis with MySQL as the database system and Apache Tomcat with Suse Linux as the Web server (see [2]). Blackberry Handhelds from T-Mobile/RIM were used as the mobile terminals. These are known for their highly developed ergonomics, integrated e-mail and telephone capability, worldwide accessibility using the GPRS network, high security standards (triple DES) and, last but not least, a freely available Java development environment with handheld simulator (see [6]).

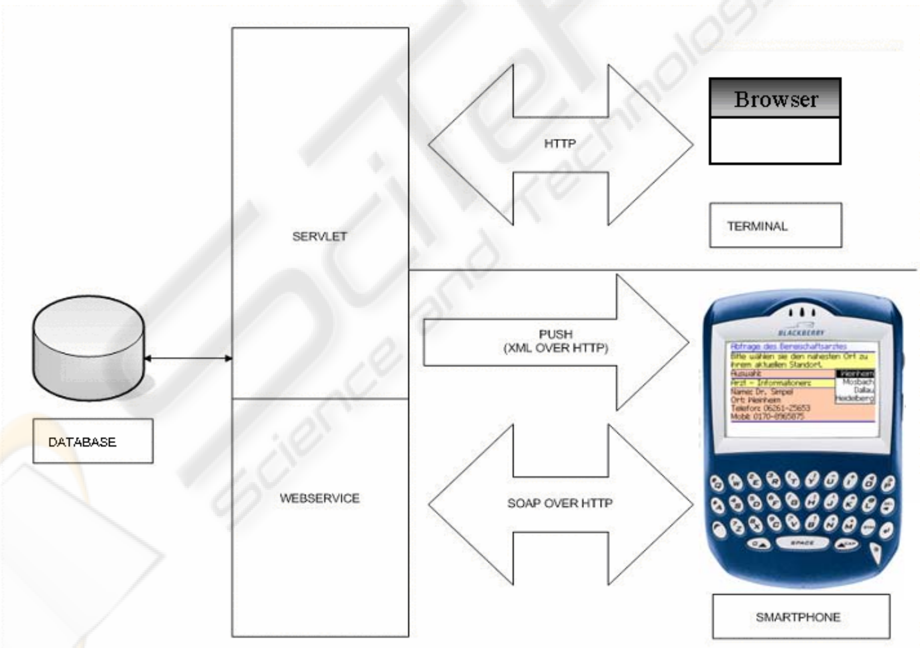

Fig. 1. The architecture of mobile applications

Another decisive reason for choosing this handheld was the option of exchanging data using the *Mobile Data Service (MDS)* with HTTP protocol in the GPRS network.

In addition to providing quick, simple data synchronization, this service enables "pushing" data from the control center to mobile terminals without the terminal having to initialize synchronization.

Each individual mobile terminal has a unique ID that must be known in order to use the "push" functions of an MDS service in access of the control center server. The MDS server provides a status code for each data record sent to a Blackberry handheld so that successful delivery of the data via the control center server can be checked. Each order sent from the control center to the mobile terminal is assigned a unique ID which allows all data to be uniquely assigned to one terminal and one order.

The mobile client makes full use of the technologies provided by the Blackberry handheld, such as multithreading, persistent storage, network communication, etc. It is becoming clear that mobile terminals like the Blackberry handheld have already attained technical and ergonomic performance that up to now was only maintained by notebooks. The Blackberry also provides integrated telephone functions that can be used directly from the applications.

Communication between the instances of the whole application is based on open standards (Web service and servlet technology): Both the control center and the mobile terminal provide Web services.

Server side Web services are available for actions initiated from the mobile devices that are connected to a servlet and that can be reached from the handheld (via MDS) using the GPRS network. Because the client also offers Web services, bidirectional communication can also take place using Web services (via MDS), which means data can be sent from the handheld and / or data can be called from the server. The connection via Web services means that the control center area and the mobile terminal are loosely linked in terms of a service-oriented architecture so that the integration of alternative mobile terminals can be accomplished easily.

Web services on the control center side are implemented in Java using Apache Axis as JAXRPC runtime, but it can be accomplished in other languages and in other runtime environments, thus guaranteeing platform independence in regard to the various control center systems used. To connect the application to an already existing control center system, one only needs to integrate the Web service defined in a WSDL file into the control center system. The Web service implementation on the mobile terminal side conforms to Java Specification Request standards (JSR) 172 [4]. The freely available reference implementation of JSR 172 by Sun Microsystems [10] was used. In order for the handheld to be able to address the Web services of the control center, a stub was generated using the WSDL description, which takes over communication with the server using HTTP in the form of SOAP messages (SOAP 1.1).

Actions that are initiated by the control center – sending messages and updating master data – are sent to the mobile device without previous query by HTTP push. The data is transferred via HTTP as the carrier protocol in an application-specific XML format. Parsing of the XML data on client site takes place via a SAX parser for mobile devices that is also included in the reference implementation of the JSR 172.

3.3 Class Landscape

The implemented classes are grouped in the categories Servlet, Web service and Client. The most important classes of each category are described as follows (figure 2).

3.3.1 Servlet

The control center system is represented by the class *CSServlet*. This class processes requests from the control center terminal and triggers the classes *XMLCreator* and *HTTPPushServer*. *XMLCreator* builds an XML-document representing application-specific data (based on user input or on database content) to be pushed to the client. This document is pushed to the mobile devices via *HTTPPushServer* instances.

Communication between *CSServlet* and the database is managed by a class *ConnectionPool* providing a pool of database connections. These are represented as instances of the class *DBCon* wrapping several java.sql-classes. Object-relational mapping is accomplished by application-specific classes. HTML on the control center side is generated by the class *PageBuilder*. Its methods are invoked by *CSServlet*.

3.3.2 Web Service

The central class *SOAPBindingImpl* contains the implementation of the methods that are invoked by the mobile device. Since simple data types can not be used as parameters in method calls (cf. JSR-172), a class *Transport* is used as a mere data container.

3.3.3 Client

MBClient is the most important class for the client-application on the mobile device. The *MBClient* instance is running permanently and only terminates if the device itself is switched off. Therefore, push-messages from the control center can be received at any time. In this case the application gets the focus and informs the user via a message-box. The class *XMLHandler* acts as a wrapper for the blackberry-specific SAX-parser. Extracted data is managed by the class *DataManager* storing it on the mobile device via object persistence of J2ME.

The source code of the class *ServiceStub* is generated from the J2ME-Wireless-Tollkit stub generator. It provides methods for Web service calls on the client side.

In addition, there are application specific classes (not shown in the class-diagrams) responsible for several tasks: Screen-classes visualize the graphical user interface, handle screen events triggerd by handheld-users and process data received from the *DataManager*. Additionally, there are classes for receiving and sending data using Web services via *ServiceStub* and for storing user input on the mobile device.

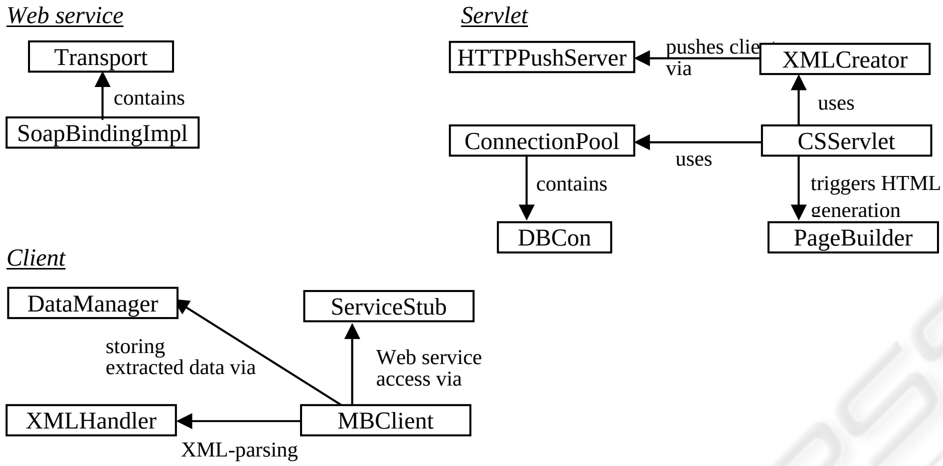

Fig. 2. The class landscape

4 Summary and Outlook

Many applications from the service industry with on-site employees offer a high potential for mobile business process optimization. They have a similar underlying structure from which a common pattern (SB2ME) can be derived for implementation of mobile solutions. Based on the technologies and open standards currently available, these types of mobile scenarios can be used and are already stable and highly available. The reference implementation introduced here can be used as a starting point for realizing mobility for more complex applications.

Based on the expected continued development of mobile technology and on the infrastructure for the integration of mobile components in existing systems, as well as the performance and ergonomics of mobile terminals, and last but not least, on the increasing acceptance of mobile technologies in the upcoming generation, we can expect that there will be more areas of application from a wide range of branches for mobile computing.

References

1. Alonso, G., Casati, F., Kuno, H., Machiraju, V.: Web Services. Springer-Verlag, Berlin Heidelberg New York (2004)
2. Deck, K.-G., Berg, M., Lichter, J., Schork, S., Stock, J., Terlinde, A., Zeller, A., Zuschlag, A.: Mobile Applications based on Web Services. In: Weghorn, H. (Ed.): Proceedings of the 1st Annual Meeting on Information Technology & Computer Science (ITCS 2004), Stuttgart (2004) 11-13

3. Die Nutzung digitaler Funktelegramme im Rettungsdienst. Zweiunddreißigster Tätigkeitsbericht des Hessischen Datenschutzbeauftragten vorgelegt zum 31.12.2003. http://www.datenschutz.hessen.de/Tb32/K18P07.htm (2003)
4. Java Specification Request 172. http://www.jcp.org/en/jsr/detail?id=172
5. Riggs, R., Taivalsaari, A., van Persuem, J., Huopaniemi, J., Patel, M., Uotila, A.: Programming Wireless Devices with the Java 2 Platform Micro Edition. 2nd Edition. Addison-Wesley (2003)
6. Research in Motion: http://www.rim.com and http://www.blackberry.com
7. Schächinger U., Röckelein, W., Kretschmer, R., Neumann, C., Nerlich, M.: Notfall-Organisations- und Arbeitshilfe (NOAH) - Informationsmanagement in der präklinischen Notfallmedizin. Notfallmedizin 27 (3), (2001) 156-160
8. Sun Microsystems: http://java.sun.com/webservices
9. Sun Microsystems: http://java.sun.com/j2me
10. Sun Microsystems: http://developers.sun.com/techtopics/mobility/apis/articles/wsa
11. Wichmann, T., Stiehler, A.: Prozesse optimieren mit Mobile Solutions, Berlecon Research Basisreport. http://www.berlecon.de/mopro