INTEGRATING MOBILE DEVICES IN WEB PROCESSES

Jürgen Dorn and Thomas Sillhengst, Electronic Commerce Competence Center (EC3), Donau-City-Straße 1, A-1220 Wien, Austria, Phone 0043-1-588801-18426 Email: juergen.dorn@ec3.at

Abstract

We present a software prototype for the communication between a mobile phone and a web-based process management system (WPMS). The communication between both is based on web services. The business process (in our example a booking process in the tourism domain) is controlled by the WPMS. The WPMS calls web services to achieve certain functionalities. The prototype allows starting a process by asking for some service from the phone. We are also able to use the phone as a server. This is demonstrated by storing user profile data in the phone and delivering this information to the web process if asked for by the WPMS and allowed by the phone’s owner.

MANAGING USER PROFILES

Today, the Internet is mainly used as an extremely large pool of knowledge. However, with Internet-based communication a number of new business models were developed in the last years. Web shops selling goods over the Internet can be open 24 hours a day and can find their customers world-wide. On one side, this large market is an opportunity for each seller but on the other side, competition is higher since the development of such a shop is cheaper than traditional shops. One possibility to be successful in this competition is to offer customers personalized services. For example, an international shop can adapt its user interface to the customer’s mother tongue.

To enable personalized services, a service provider needs customizable products and moreover, he requires information about his customers. This information may be obtained by simply asking customers with web forms about their preferences, by deducing preferences from past transactions or by some intelligent reasoning about the customers’ behaviour in the web shop. A service provider may use an information system architecture including a User Profile Management System (UPMS) as in the following figure.

Figure 1: Simple information system architecture for user profile management

This architecture is sufficient for the service provider if the customer supplies him all required data. A customer, however, does not buy only in one web shop. The advantage of web shops for the customer is the easy comparison of products in different shops. Thus the customer fills out web forms quite regular. Several user agents such as Internet browsers already support users by storing data for certain fields of forms. We may store our name or a password to facilitate the completion of forms. This procedure is adequate if only few data is required. If, however, preferences about complex products are required, such forms are insufficient. A client agent should have reasonable knowledge about his user and may be authorized to negotiate with the service provider’s UPMS.

Our application domain is tourism. Here complex holiday packages shall be configured. There may be a portal such as www.tiscover.at that recommends several services for a holiday. If the user agent supplies the service provider with data, we must consider security and privacy issues. We only want to give away data that is really required and moreover, we must secure that delivered data is not misused. These problems shall be addressed by the platform for privacy preferences (P3P) standard [5] an extension of Netscape’s open profiling standard. P3P defines an XML-based language to exchange profile data between a client and a server. P3P proposes that web sites (the server or the UPMS) publish their policy, what they do with obtained data, in a standardized format. The user agent can then compare this policy with user’s preferences and will only submit that data that is accepted by the user.

Customers in the tourism domain may use personal computers to plan their holiday, but during holiday they usually do not have access to a PC with their user profile. If they have access to the Internet, there would be the possibility to store a profile on a server. The second possibility for owners of sophisticated mobile phones is to store the profile in the phone. If the user searches for some information or wants to book an additional feature during his holiday, the phone may negotiate the required data with the service provider.
We have developed a prototype software for a mobile phone, that stores user preferences and can exchange data with a WPMS, considering the user’s preferences and data privacy issues. In the following section we describe how processes are modelled and executed by a WPMS. Then we describe the storage and exchange of user data. In the third section we describe technical issues of the prototype and conclude with some remarks on future expectations.

BUSINESS PROCESS MODELING AND EXECUTION

The MOVE project (www.ec.at) aims at an integrated development environment for virtual enterprises. One of the most important functions in a virtual enterprise is the communication between partners and the customers. First step in the development process is the definition of the desired business model. The second step is the definition of business processes such as for example “booking a holiday for a customer”. Business processes may be defined hierarchically. On the lowest process level, we have technology driven processes such as “exchanging data with mobile phone”. Whereas the high level processes are very specific for a new business model, the low level processes can be reused easily by parametrizing. For our example, we must use specific data that is exchanged between process and phone.

A process model may contain several activities to be performed and a control flow describing the choreography of the process. An activity may be a sub process, a web service call or some other action performed either by another program or by a human. Our main focus in the project, however, is on a web services [8] based architecture. Besides the control flow, the information flow is specified by defining data records and flow between activities. In the data view, we specify the information flow and the data which is queried from the customer. The business model and the hierarchically defined processes are modelled with ProSpec a Java-based graphic process modelling tool [6].

The modelled process can be stored as an XPDL-process. XPDL [9] is a XML-based language proposed by the Workflow Management Coalition. XPDL enables to describe the control flow in a process and can be easily extended to integrate for example also web services. OBE (Open Business Engine) [2] is a workflow management system that executes and controls processes specified in XPDL. We have extended OBE by web service functionality enabling us to execute XPDL processes. The process itself can be deployed as a web service. The WPMS is used to execute and control a booking process for a tourist. In the prototype scenario, a booking process is deployed as web service that can now be initiated by a mobile phone simply by calling the web service with appropriate parameters.

USER PROFILE REPRESENTATION AND EXCHANGE

We have developed a prototype for a mobile communication in the described environment [7]. This prototype is implemented with the Java version for mobile devices (J2ME). The program allows calling a web service from the phone thus starting a process controlled by our WPMS (e.g. a search for a hotel). Moreover, the prototype realizes a small web server in a mobile phone that can be called via the SOAP-interface and with that as a web service. A user can login in a usual web shop via his PC. If the service provider requires certain data it may query the phone if the number of the phone is known. If the web server needs data from the client, it sends a profile template to the client. The client software looks into its database whether the desired attributes are defined. If not, the user is asked to supply the data to the phone. If attributes are defined, the client investigates the policy of the server to decide whether the data can be transferred to the web server. This decision is based on the users preferences stored also in the database of the phone. The data exchange is based on XML-schema. The following code shows exemplary an abbreviated exchanged profile:

Figure 2: Improved architecture with negotiation between user agent and UPMS
The user can decide for each attribute whether it may be sent automatically if asked for. Thus a user may specify which data may be transferred automatically to a tourism portal. However, if credit card information is required the user may want to be asked again and may accept the transfer with a single key stroke on the phone. For other attributes the user may specify that they shall never be transferred to a tourism portal.

Usually the user will not specify his preferences immediately with his phone’s user interface. Therefore, also a profile portal is required, where the user will specify his preferences and then load it on his phone. Such a portal may be provided by mobile phone operators. The software for the mobile phone can easily be downloaded from standard web sites or from special web portals for mobile devices.

**IMPLEMENTATION**

For the implementation of the prototype software on the mobile phone, there are four aspects significant: security, privacy, user-friendliness and performance. First, the security is absolutely necessary for handling and transmission of sensible data. The Mobile Information Device Profile 2.0 (MIDP2.0) of J2ME, a subset of java libraries, is designed especially for mobile devices and offers all necessary techniques for secure networking and secure storing of data. Privacy is addressed by P3P. User-friendliness should be guaranteed by a simple interface with easy navigation. For an easy handling of the application most important system parameters can also be set remotely from the service provider. So the user will not become annoyed by inserting long internet addresses in the mobile phone. Performance is also important for user convenience. We have no influence on the network performance, which depends on the amount of transferred data and used protocols, but we tried to focus on fast processing of data on the mobile phone. Therefore thin libraries and parsers which can work in mobile devices with limited hardware resources were used for the prototype.

The program consists of ten modules for the main tasks. The advantage of this modularisation is that the software can be adapted from an administration tool for user profiles to an administration tool of any content with communication possibilities to any Internet-capable device.

![Modules of mobile phone software](image)

**Figure 3:** Modules of mobile phone software

The modules at the first line of figure 3 are for the administration of user data. User profiles can be saved and loaded on the mobile phone. Two communication types are available for the transmission of user profiles and other data. Sockets enable a connection oriented communication and datagrams a connectionless network communication. Profiles can be edited on the mobile device. However, the user cannot add new attributes. The mobile phone can receive profile templates developed on the server side. They describe, like a form, the structure and data types of user data. kSOAP and kXML are XML parsers developed from Enhydra [2] for devices with low hardware capacities. Both parsers are used for the internal parsing of SOAP requests and XML user profiles. Furthermore, a syn-
chronization of profiles with a profile portal is integrated in this software. A user can adjust settings of the software. This can also be done remotely from the service provider (e.g. by a SMS Message).

The communication between mobile phone and WPMS is in principle a client-server communication. The mobile phone starts the service of the WPMS by sending a request (e.g. a request for a holiday offer). The request includes the preferred start and end time of the vacation. Since it is not known in advance how long the WPMS needs to generate a holiday offer, the mobile phone software runs in server mode. Java provides an internal registry for pushing applications remotely so that neither the application must be active nor the user has to have his mobile phone switched on. If the WPMS requires additional information about the user, it sends a profile template to the user and the user agent will send back the requested attributes, according to his configuration. When the WPMS has enough information it creates the individual holiday offers and sends them back to the user, who can watch it on the mobile phone.

CONCLUSIONS

We have developed a software prototype for a flexible communication between a mobile phone and a web-based process management system that may control business processes in an e-business portal. The software is developed as an application framework enabling easy extensions or changes. This flexibility is required since standards as well as the used hardware is changing continuously.

Only recently a first phone realizing the Java MIDP 2.0 is on the market. Current phones have restricted resources which advice that not the full P3P proposal is implemented in the phone. Extensive and flexible XML-structures require considerable memory space as well as processor speed. However, we expect improved hardware for the next years allowing the full functionality without great effort.

An open question is how end users accept such a smart service on their phone. Considerable benefits must be offered to make them aware and conscious about the possibilities. In a new project we are trying to integrate the proposed communication in a solution for a conference management system [1]. Here customers are abroad and do not have their typical work environment at their side. Information on current sessions and talks as well as information on local cultural, transportation and other services on demand and as a push service may propose such a benefit.

At the moment, we do not have implemented the full P3P standard. Moreover, we must develop this also on the server side (e.g. for a conference portal). Considerations must also be taken whether our solution based on P3P policies conforms with European and international regulations on security and privacy. Since P3P focuses on a browsing session with an Internet browser and not on ad-hoc data exchange in business processes other requirements for privacy may be important.

Finally, the developed software seems to be so generic that it should also support the transmission of other data formats (not only XML-data). For the conference portal we consider, for example, the transmissions of conference talks via UMTS. This transmission is dependent on several aspects modelled in the portal. Modelled business processes, a web-based process management system and a web service based communication process facilitate a generation of such transmissions.

LITERATURE