MOBILE APPLICATIONS AND SERVICE-ORIENTED ARCHITECTURE INTEGRATION

A Practical Approach in Incorporating Mobility with Enterprise Information System

Guillaume Koum, Augustin Yekel
Ecole Nationale Supérieure Polytechnique, Université de Yaoundé,P.O. Box. 8390 Yaoundé, Cameroon

Tam Sangbong, Raoul M. Tsamo, Arnaud Fosso
Soft-Tech International Inc, P.O. Box. 15412 Douala, Cameroon

Keywords: Mobile Applications, Service-Oriented Architecture, Web Service, Mediator, Gateway, Computer Reservations System, Java 2 Mobile Edition.

Abstract: Mobile Applications and Service-Oriented Architecture (SOA) are two aspects of Computer Science which offer to Enterprise Information Systems (EIS) more agility and flexibility in order to follow the constant evolution of enterprises business. However, the inadequacy of hardware resources and wireless network availability and bandwidth constitutes a major constraint when developing any mobile application. The challenge is exacerbated for developers when trying to integrate these applications with EIS which are naturally heavier. This document presents this integration in the context of an EIS built on SOA. Added to the basic tools needed for local data storage and communication with the server, a mediator (on the device) is introduced to facilitate interactions between the application and the service. We also study reasons which lead us to the choice of implementing a gateway on the server, in order to facilitate the service consumption.

1 INTRODUCTION

The mobile terminal has long served as a communication tool between people. Its use was further diversified with the implementation of gaming, organizer and calculating applications. This evolution reached a critical point when it was necessary to allow these applications running on the mobile to be able to communicate with external applications, thus granting real mobility to the user.
The object of our work is the integration of the mobile applications in the global system of the enterprise, notably in the setting of a SOA.

In this document we present the architecture of mobile in section 2 which also presents how the problem of client (on the mobile) and server (backend system) communication is tackled. After the description of SOA applications in section 3, section 4 will present how they are integrated with mobile applications. Before the concluding remarks section 5 and 6 respectively show the description of our prototype and related work in this area.

2 ARCHITECTURE OF MOBILE APPLICATIONS

Like for other applications describing mobile applications architecture refers to presenting their various building blocks and the models which govern interactions between the blocks. However, mobile applications are specific primarily because mobility.
Mobility, the hardware limitation of mobile devices as well as the fact that devices need to communicate with backend systems, really influence mobile applications architecture.

Koum G., Yekel A., Sangbong T., M. Tsamo R. and Fosso A. (2007).
MOBILE APPLICATIONS AND SERVICE-ORIENTED ARCHITECTURE INTEGRATION - A Practical Approach in Incorporating Mobility with Enterprise Information System.
In *Proceedings of the Second International Conference on e-Business*, pages 204-208
DOI: 10.5220/0002113802040208

2.1 Types of Mobile Application

The first criterion of distinction between mobile applications is the application connectivity mode.

Then like other applications they can be layered from three different perspectives (user interface, processing, and Data layers). The layered view of the application is applied to another mobile applications differentiation according to the client/server paradigm: the thin client and the fat or smart client.

The fat client solves most of thin client disadvantage and improves performance by allowing data and operations to be executed locally. It facilitates the interaction with other applications on the terminal and can be integrated to the existing infrastructures. The inconvenience is that it requires more resources (Juntao Y., Michael, 2004).

2.2 Client/Server Communications

The fact that mobile applications are mostly partially disconnected imposes constraints on the management of data storage and transfer between the client application and the enterprise server. That is why a communication model or data exchange protocol needs to be define thoroughly. The low debit and the occasional unavailability of the wireless network can impede the normal execution of the application if it interacts with the server in synchronous mode. So, the use of an asynchronous communication model is better indicated for mobile applications.

However, asynchronous communication poses the problem of data synchronization. This affects most partially-connected applications which use the *store - and - forward* model. (Jourdain, Sébastien, 2002)

To execute the synchronization it is necessary to implement solutions such as the protocol SyncML (Crochet-Damais, Antoine, June 2004).

Along with synchronization, another important facet of the problem of data transfer between the client and the server is the format of the data to be transmitted. Because XML (eXtensible Markup Language) data is generally “large” (Enrique Ertiz., April 2004), WBXML (WAP Binary XML) which is targeted to mobiles terminals, is preferred. (WIKIMEDIA Project, January 2006).

We will now examine the architecture of service-oriented application before looking at its integration with mobile application.

3 SERVICE ORIENTED ARCHITECTURES (SOA)

Service oriented architecture is defined as a set of methods, practices and environments that allow functionalities of an application to be provided and used as a set of services at a suitable level of detail for the consumer of the service. Services can be invoked, published, and outlined and are separated from the implementation by a unique and standardized interface (Gartner Group, April 2005).

A SOA is centred on a principal concept: the service which is the means by which a supplier satisfies a consumer's needs.

The SOA puts in place a certain number of entities that have to interact in order to permit a comfortable use of the functionalities of the application by means of services (the service user; the service supplier, the service directory, the service contract, the proxy, communication and transport protocols).

Web services are currently the most widely implemented service oriented architecture (Keen, Martin, 2004).

More details on mobile and service oriented applications architecture can be found in references.

4 INTEGRATION OF A MOBILE APPLICATION WITH A SOA

4.1 Principles

We present here the integration of a mobile application on a server whose architecture is service-oriented. The goal is to enable the application to integrate the SOA while taking into consideration the fact that the connection is not guaranteed at all times and that capacities of the mobile terminal are limited.

The mediators’ model (Kureshi, Arif, 2004) has been used to allow the application to reach the offered services in order to preserve a weak coupling between the various aspects of the application and the service. Therefore the mediator will execute the operations normally reserved for the services. It has the following roles:

- Intermediary between the application and the service: It can offer the application to do in several small actions of operations that the service asks to make in only one time;
- Local processing unit: it encapsulates processes and local data which enables,

among others, the application to be able to function in a partially connected environment;
- Scheduling: it must be able to schedule the resumption of transactions. For this, it needs to be autonomous;

4.1.1 Mediator–service Association

A mediator can be the mediator of several services if these have a narrow relation between them. It is the case if transactions exist in which these services participate explicitly from the point of view of the mobile application.

4.1.2 Communication between Mediators

It is possible that mediators need to use data used by another mediator. To maintain the weakest possible coupling between the different mediators as is maintained between services themselves, one mediator should not directly address another. An object of the application must be able to take data from one and transfer it to the other.

4.2 Proposed Architecture

The model presented above is summarised on the figure 2 and explained in subsections below.

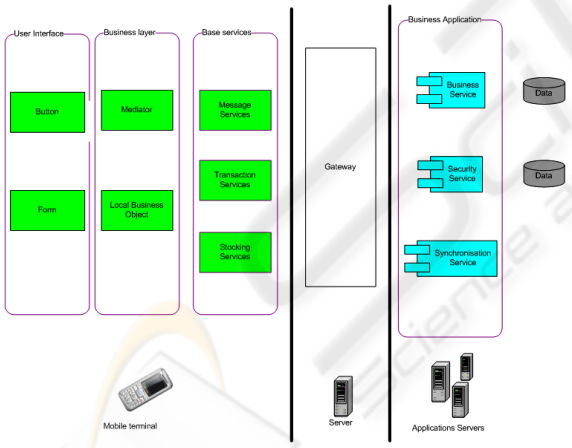

Figure 1: Proposed architecture.

4.2.1 Client Side (Mobile Terminal)

- Basic infrastructure: The basic infrastructure comprises the set of services enabling the storage of data on the mobile terminal and communication network operations. The topic is not about services offered by the SOA but of tools that can be implemented by the developer or placed at his disposal by the development environment or external libraries;
- Business logic layer: This layer contains mediators and possibly entities executing other processes that are not provided by the services;
- User Interface: The user-interface comprises all the components enabling the interaction with the user. These elements can be visual or vocal;

4.2.2 Server Side

- Service - oriented architecture: We developed this work on the basis that the application server has a service - oriented architecture, which means that functionalities are distributed in the form of services that can eventually be integrated on an enterprise services bus;
- Gateway: We presented the different properties of this gateway. It has an optional role, depending on the capacities of the mobile terminal and the network on which the application is going to function.

5 IMPLEMENTATION OF A FLIGHT INFORMATION APPLICATION

This section applies the studied concepts in order to verify their validity. Thus, we have developed a working prototype, the Flight Info application. The system provides information on scheduled flights of various airline companies. The importance of a flight information application can be recognized by anybody having had to plan a long distance journey requiring several stopovers.

5.1 General Description of the Use of the Product

The product will be used by the commercial agents of travel agencies or any other third parties. The software comprises two main modules, one for agents of travel agencies and the other for travellers.

5.2 The Different Functionalities

- Obtain flight information (costs, timetables, number of available seats, etc.);

- Determine flight itineraries while specifying constraints (minimum cost, timetables, stopovers, etc.)
- Book customer reservations (card number can be stored since data is isolated);
- Receive and send notifications from and to the travel agency;

5.3 Application Architecture

The application architecture is the one that we proposed above. We describe the different services and mediators who build the application server and the client application respectively.

5.3.1 The Application Server

It comprises the set of services that offer Flight Info
The identified services are:

- The information service on flights and accommodation (hotels); offers customers flight and hotel information
- The booking service; enables the customer to book a seat on a flight or a room in a hotel. This service uses the electronic payment service for confirmation;
- The electronic payment service: enables electronic payment of a booking;
- The customer management service: enables registration, modification or deletion of customer information.
- The agent management service: enables registration, modification or deletion of agent information;
- The security service, assures the security of the application (authentication of users, etc.)

5.3.2 The client Application on the Mobile Terminal

It comprises all functionalities that are accessible from the mobile terminal (client of the travel agent or the traveller). We have the different mediators and services to which they correspond

- The information and booking mediator: this is the mediator between the application and the information service on flights and accommodation and the booking service;
- The customer management mediator: this is the mediator between the application and the customer management service. It will enable access and modification of customer data;
- The agent management mediator: this mediator enables the agent management service, notably for the access and the modification of information on this agent;
- The notification management mediator, this mediator permits to manage the sending and reception of notifications.

5.3.3 The Gateway between the Client Application and the Application Server

It is relevant mainly for the functioning of the flight itineraries search service. It permits to store the important data in order to send only a tolerable amount to the application.

5.4 Tools and Technologies used to develop a flight Information System

Several tools and technologies were used to develop the prototype. We describe then below.

5.4.1 J2ME and J2EE

J2ME is a Java (language) platform used for developing mobile applications. The framework provides three main categories of virtual machines: the CVM (C-Virtual Machine) which implements nearly all specifications of the regular JVM but is optimized for the mobile; the KVM (Kilobyte Virtual Machine): a very light version of the JVM (40 - 80 KB) which supports fewer functionalities; the CardVM (Card Virtual Machine): a version of the virtual machine for applications on SIM cards.

5.4.2 Simplicity

Simplicity for Mobile Servers is a development environment especially for mobile Client–Server applications. This environment permits to develop enterprise servlets and integrates MIDPS that permit to develop Midlets applications. It offers some key components:

- The XML Export component allows to quasi-automatically export captured data in XML by a servlet. This module proves to be very useful for the communication between the client and the server.
- The Transaction module is available at the level of the client Midlet application. It permits to send requests to a servlet via HTTP and to receive the data in XML format

5.4.3 JESS

JESS (Java Expert System Shell) is an inference engine written in Java. It enables inferences based on rules and conditions defined in a script language. It permits mainly forward reasoning, but backward reasoning is possible, both through an algorithm called RETE. We used JESS to search for itineraries between cities.

6 RELATED WORK

Found in the literature are the works of (Do Van Thanh; Jorstad, I., July 2005).and (Duda, I.; Aleksy, M.; Butter, T., July 2005).

7 CONCLUSIONS

Mobile applications are an opportunity for enterprises that wish to cover wide areas and at the same time be closer to the user. However, developers face challenges because of a certain number of restrictions in term of material and network infrastructure. We have examined in this paper how these problems can be solved in the case where the mobile application integrates an enterprise application with a service-oriented architecture. This work permitted us to propose an architecture allowing the mobile application to access services through the intermediary of mediators. However, two important aspects must be studied in depth: security and the supply of services.

REFERENCES

Anderson, Christoffer, 2004. *GPRS and 3G Wireless Applications, The Ultimate Guide to Maximizing Mobile Internet Technologies*, Wiley Computer Publishing.

Jourdain, Sébastien, 2002. *SAMS : Environnements coopératifs Synchrone, Asynchrone, Multi-Synchrone pour les équipes virtuelles*, Mémoire de DEA, Université Henri Poincaré.

Juntao Y., Michael, 2004. *Enterprise J2ME, Developing Mobile Java Applications*, Prentice Hall PTR.

Lee, Valentino, 2004. *Mobile Applications, Architecture, Design, and Development*, Prentice Hall & Hewlett Packard.

Livingstone, Dan, 2002. *Advanced SOAP for Web Development*, Prentice Hall.

Mbassi, Cyrille, 2005.*Conception et mise en oeuvre d'une architecture orientée service*. Master of engineering's end of course thesis in computer science National Advanced Polytechnic School. Cameroon.

Fosso, Arnaud, 2006. *Intégration d'applications mobiles dans une architecture orientée service*. Master of engineering's end of course thesis in computer science National Advanced Polytechnic School. Cameroon.

Crochet-Damais, Antoine, June 2004. *Panorama des outils de synchroniation client / serveur*. http://solutions.journaldunet.com/0406/040601_panorama_synchronisation.shtml. JDN Solutions.

Enrique Ertiz, C., April 2004. *Introduction to J2ME Web Services*. http://developpers.sun.com/techtopics/mobility/apis/articles/wsa. Sun Publications.

Gartner Group, April 2005. *Architecture and Planning for Modern Application Styles*. www.gartnerroup.com. Gartner Group.

Hemphill, David, July 2002. *Using J2EE to drive J2ME Applications*. http://www.theserverside.com/articles.

Juntao, Michael, June 2002. *Securing your J2ME/MIDP Applications*. http://www-128.ibm.com/developerworks/library/j-midpds.html. IBM Publications.

Keen, Martin, 2004. *Patterns: Implementing an SOA using an Enterprise Service Bus*. http:// www.ibm.com/redbooks. IBM Publications.

Kureshi, Arif, 2004. *Architecting Disconnected Applications using Service-Oriented Architecture*. http://msdn.microsoft.com/library/enus/dnppcgen/html/develop_disconnect_mob_apps.asp?frame=true#develop_disconnect_mob_apps_topic2. MSDN Publications.

Stemberg, Scott, October 2001. *Syncing Data, an introduction to SyncML*. http://www-128.ibm.com/developperworks/xml/library/wi-syncml/.

Sun Microsystems, May 2001. *L Over-The-Air User Initiated Provisioning Recommended Practice for Mobile Information Device Profile 1.0*. http://java.sun.com/products/midp/OTAProvisioning-1.0.pdf. Sun Publications.

Warren, Nigel, June 2005. *Talking Service-Oriented Architecture Mobile.* http://today.java.net/pub/a/today/2005/08/02/mobile1.html. Java.net Publications.

Wikimedia Project, January 2006. *XML: Managing Data Exchange/SyncML*. http://en.wikibooks.org/wiki/XML:_Managing_Data_Exchange/SyncML. Wikimedia Project.

Do Van Thanh; Jorstad, I., July 2005. A service-oriented architecture framework for mobile services. *Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference/ E-Learning on Telecommunications Workshop. AICT/SAPIR/ELETE 2005*. P 65 - 70.

Duda, I.; Aleksy, M.; Butter, T., July 2005. Architectures for mobile device integration into service-oriented architectures. *ICMB 2005, International Conference on Mobile Business*. P 193 - 198.