Mobile Devices and Web Services

TOMAS KOZEL, ANTONIN SLABY
Univerzity of Hradec Kralove
Department of Informatics and Quantitative methods
Rokitanskeho 62, 500 03 Hradec Kralove
Czech Republic

Abstract: Service oriented architecture (SOA) presents one of modern approaches used in the process of information systems development . One of the technologies the SOA is based on are web services. Rapid development of capabilities of mobile devices (including PDAs, Smart phones or mobile telephones) as well as mobile networks opens the possibility of mobile devices to access the web services too. In this contribution there will be described selected approaches as well as models used in the process of implementation of a mobile client of information systems mainly those which make use of web services.

Key–Words: mobile device, java, Java ME, SOA, web service, WSA, kSOAP, BlackBerry

1 Introduction

At present mobile devices undergo rapid development which results in longer performance without charging, better performance larges set of functions implemented for the use of customers and consequently the area of their potential use spreads constantly. Similar progress can be seen and recognized in the area of mobile communication networks mainly in the area of mobile data services. Smart phones start to take control of the market of pocket computers and communicators and the classical PDAs withdraw and disappear. The majority of mobile telephones recently supports mobile Java (Java ME), many communicators possess their own operating system (Symbian, Palm OS or Windows Mobile). All these platforms enable relatively easy opportunity to make mobile applications including data communication access. In connection with further modern trend in the area of management and software development - i.e. transition to service oriented architecture, there raises possibility to use the mobile devices as clients of these services. Using mobile clients brings quite new views and approaches in many business processes. Permanent access to up-to-date data of business information systems from any place is useful in many areas such as selling products (B2B, B2C), relations to the customer (CRM) etc.

In the further text there will be described selected attitudes and subsequently technologies which are applicable for the implementation of the client of information system (resp. of web service) on the mobile device. First will be mentioned solutions based on mobile Java technology (J2ME) and then the solution offered for BlackBerry® platform developed by the firm Research In Motion (RIM).

2 Mobile attitude to information systems

Mobile devices that are equipped with the mobile edition of java virtual machine (Java ME) are able to set up communication in the mobile data network making use of the HTTP protocol and selected models can use general socket connection. This capability is enabled for version MIDP 1.0 and later versions. The latest versions of mobile java can offer a row of application program interfaces (API), that significantly spread the capability of mobile applications. These APIs include also an interface for the access to the web services (WSA) [7]. In the following text there will be briefly described variants of the access to information systems which are enabled by these technologies.

2.1 Proprietary solution of the communication

Classical solution of the problem of the access to the selected parts of information systems from mobile device can be realized either by classical socket connection and economic proprietary protocol or through the common HTTP(S) protocol. In both cases there is necessary to implement connector on the side of information system (or use a web server). The connector enables the mobile device the access to the selected functions and data of information system. This solu-

tion is more suitable for very specific problems and less suitable for more common problems. Client is often in this case burdened by quantity of application logic, that it has to perform. This fact naturally reduces performance of the device. The further disadvantage is the necessity to create one purpose connector which is dedicated for connecting the mobile device only. On the other hand this solution has the advantage of economic data communication which at the same time makes the data transmission between client and information system quicker and cheaper. The model of this solution is depicted on figure 1.

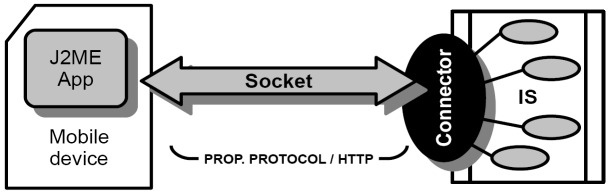

Figure 1: Proprietary solution

2.2 Making use of Web Services

Calling of remote objects through the web services (WS) brings a lot of new possibilities and opportunities into the area of the distributed applications development. Some of them address the problem of creation of applications for mobile devices. Let us mention these facts:

- Computationally arduous tasks can be performed completely on the side of IS (application server).
- The web services are accessible from various platforms and consequently there is not necessary to implemented solution dedicated for the access from the mobile device only.
- The solution is easier to maintain and modify as the changes mainly concern business logic of application server only and so there is not necessary to distribute the changes to all mobile clients.

On the other hand there is commonly known that the SOAP protocol used for the communication with the web service, which belongs to the family of XML protocols, is not very economic way for the data transmission. Two main disadvantages of the solution come out from this fact:

- A large amount of data is transmitted which makes the transmission process slower and may lead to raising of its expenses.
- The parsing of an XML document in the mobile device is computationally more requiring and may lead to slowing of the application.

2.3 The combination of WS and the economic transmission

The combination of previous two attitudes may lead to the further model. The access to the selected functions of the IS will be enabled via the web services as it is described in part 2.2. Mobile client but will not connect to these web services directly through the SOAP protocol but it will use specially for this purpose designed gateway and it will communicate with this gate via an economic compact protocol in a similar way to the one mentioned in 2.1. The gateway then organizes all the communication with the web services. The principle is depicted on figure 2.

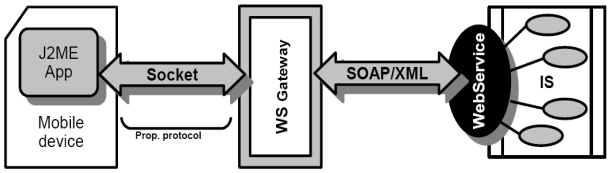

Figure 2: The gateway for connecting mobile device to web service

3 Exploitable technologies

Here will be described possible technologies for realization of above mentioned variants for the communications of the mobile device with IS. Due to less commonness of the first mentioned model 2.1 this variant will not be discussed in details.

3.1 kSOAP

kSOAP [8] is a library which enables to create application in the mobile Java that communicates directly with the web service and uses the SOAP protocol. One of the basic advantages of kSOAP is the fact that it can be used also with older devices which uses the older MIDP1 profile and also with devices that do not offer the WSA support (see 3.2) . It is installed in the form of an additional library for the mobile application in the mobile device. The library implements basic classes for the cooperation with XML and SOAP protocol, it does not use static typing and therefore it is necessary to use either a HashMap or implement `KVMSerializable` interface to access structured user types. At present there is available the second

```
try {
  //SOAP object specifiing the namespace
      and the method
  SoapObject so = new
      SoapObject("http://ws.crm/",
      "getCustomer");
  //parameter setting
  so.addProperty("index",Integer.valueOf(
      tfCustNo.getString()));
  SoapSerializationEnvelope env = new
      SoapSerializationEnvelope(
      SoapEnvelope.VER11);
  env.setOutputSoapObject(so);
  HttpTransport transport = new
      HttpTransport(
      "http://my:8084/CRM_WS/CRMService" );
  //method call
  transport.call("",env);
  //retrieving the result value
  SoapObject res=(SoapObject)
      env.getResponse();
  striName.setText(
      res.getProperty("name").toString());
  striSurname.setText(
      res.getProperty("surname")
      .toString());
} catch (Exception ex) {
  ex.printstackTrace();
}
```

Sample 1: kSOAP example

revised version of the library. The documentation accompanying the library but seems very incomplete. Sample code 1 shows one simple example of using this library.

3.2 J2ME Web services APIs (WSA)

Specification JSR-172 [7] often referred as WSA (Web services API) is other possibility of connecting the mobile client with the web service. The purpose of the specification is to enable SOAP/XML calls to the web services in mobile Java and it enables some basic parsing of XML documents too. The implementation of this API by the mobile device is the necessary condition that enables the access to the web services on this mobile client. There is possible to use a stub generator that generates a set of classes needed for the access to a web service. It enables to call methods of the remote service on the client side as it is a local object. The generator is the part of all newer versions of Java ME toolkits. It also creates all the necessary user types of the service from given WSDL document. The classes generated in such a way can be simply included into mobile application and used. The example of code showing communication with web service using stub is in the sample 2. Using of the stub results in the fact that there is not visible any trace of communication with remote web service.

```
//Creating stub instance
CRMService srv = new CRMService_Stub();
Customer z;
try {
  //Service call using stub instance
  z = srv.getCustomer( Integer.parseInt(
      tfCustNo.getString()));
  striName.setText(z.getName());
  striSurname.setText(z.getSurname());
} catch (Exception ex) {
  ex.printStackTrace();
}
```

Sample 2: WSA (JSR-172) example

3.3 RIM BlackBerry Solution

Canadian company Research in Motion (RIM) offers one interesting possibility of access to the web services by means of BlackBerry services. The customer of the service, offered by some mobile operators, can create or partially develop the mobile client in a free of charge development environment BlackBerry MDS Studio [9]. Such developed mobile client possesses the ability to communicate remotely with the web services and/or with selected database severs. In addition to it the mobile device works virtually as a terminal of inner company network due to features of BlackBerry service. In such a way there is ensured high level of security of communication between the mobile device and the web service and at the same time there is fully centralized administration of all the mobile devices (which include installation, application maintenance, security policy administration etc). Running of the client application requires the previous installation of so called MSD Runtime which communicates via firm Blackberry enterprise Server (BES) with intended web services. The communication between Blackberry device and BES is realized by a simple and compact protocol. The situation is depicted in the figure 3.

The development process starts again with the WSDL document of the web service which enables to create either objects of messages and accompanying data types only or even GUI forms for the applications. Event driven application logic on the client side is implemented in JavaScript. MDS applications can also respond to so called push messages sent by the server thanks to implementation of WS- Eventing standard. The resulting finished application is distributed into devices via central MDS and BES

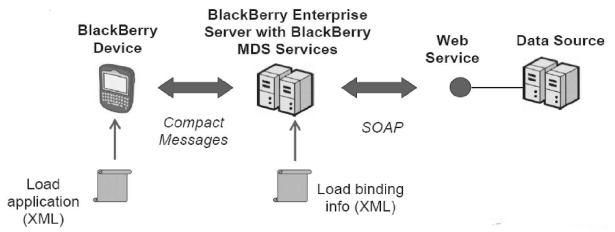

Figure 3: Architecture of RIM BlackBerry MDS Studio [9]

Servers.

4 Case study

In this part of the contribution there will be introduced the project of B2B mobile client BBShop which served for testing various capabilities and the stability of solutions based on BlackBerry services. The aim of the study was creation of mobile application for travelling salesmen who directly realize orders for their (stable) customers in any place. The application offers the possibility of selection of the customer, browsing contracts, remarks made about and for customers, searching, selection from the catalogue using advanced filtering, assuming price reductions, making orders from the catalogue, or by means of Bluetooth reader of barcode, dispatching the order and an immediate confirmation of the order and its options that include quantity, reservation etc. Everything is done via BlackBerry device. All the operations were implemented as the web services and the kernel of the application logic was executed on server due to restricted amount of memory of mobile devices and with respect to the rapidly changing catalogue. For the implementation of the application server there were used the following technologies Apache Tomcat, Hibernate, Spring (Core+WS), JibX, Apache Axis, Quarz). To reduce the workload of the enterprise IS there is established synchronization of basic tables of keys (catalogue, customers, etc.) once a day. The only information which is exchanged between information systems and application server in real time are the orders and their confirmation. Figure 4 gives a sample of GUI of the client of the application BBShop.

In spite of initial worries about the performance - mainly time response and efficiency of transmission in the GSM network - the application is surprisingly quick and stable. In connection with the accessing of the large-scale catalogue it was necessary to implement a catalogue pagination and the advanced filtering to prevent transmission of large portions of data since BlackBerry technology enables transmission of

Figure 4: Client of BBShop application

data messages up to 32 kB only. Using the BlackBerry client also helped to solve the former problem of raising load of the information system in late afternoon and early evening hours. In this period of time all the salesmen used to dispatch orders from customers which had been collected during the actual day. BBshop enabled to distribute the application load of the information system more proportionally during the day and at the same time achieved very quick distribution of orders. BlackBerry based solution is but substantially more expensive than solutions based on other in this contribution mentioned technologies and is suitable mainly for larger companies, that use or intend to use BlackBerry as their basic communicator.

5 Conclusion

At present mobile devices already have sufficient computational performance to maintain communication via the SOAP protocol. There exist reliable technologies, that enable implementation of mobile client of the web services. As to rate of transmission of the data so far prevails the variant of solution described in part 2.3. of the contribution with compact and economic protocol that is implemented for example by BlackBerry services. The authors intend to realize more exact measurement and testing in further stages of the research.

One of the essential and here not mentioned aspects accompanying the communication of mobile clients with web service is the problem of security and safety of both transmission channel and the web service itself. None of here mentioned solutions have built-in support of the protocol WS-Security. WSA and kSOAP could be supplemented with the encrypted protocol HTTPS and with authentication of users. Easier situation is in case of BlackBerry ap-

plication as here secure and authentication possessing access is the organic part of the service itself and mobile device is assumed as computer of inner network. Even in this case there is but possible to make use of HTTPS and authentication as additional way for security of communication between BES Server and web service in inner network.

The results of the project BBShop mentioned in our case study indicate that use of web services raise and widen possibility of exploitation of the mobile telephone (or communicator) to the much higher level. Factors that limit or restrict this technology are so far relatively slow commonly offered data services for the mobile phones and of course capabilities of the mobile devices. There is reasonable to implement mobile client for devices having reasonably large display and suitable way of governing input (pen, QWERTY keyboard).

References:

[1] Giguere, E. Service-Oriented Architecture and Java ME.
URL: http://developers.sun.com/mobility/midp/ttips/soaintro/ [online], *cit. 24.8.2007*

[2] Giguere, E. SOA Without SOAP: The Java ME Perspective.
URL: http://developers.sun.com/mobility/midp/ttips/soanosoap/ [online], *cit. 24.8.2007*

[3] Luqun Li, Minglu Li, Xianguo Cui.The Study on Mobile Phone-Oriented Application Integration Technology of Web Services, Lecture Notes in Computer Science, Volume 3032/2004, Springer, 2004, ISBN 978-3-540-21988-0

[4] Oritz, E. Understanding the Web Services Subset API for Java ME,
URL: http://developers.sun.com/mobility/midp/articles/webservices/ [online], *cit 1.9.2007*

[5] Tergujeff, R., Haajanen, J., Leppanen, J., Toivonen, S. Mobile SOA: Service Orientation on Lightweight Mobile Devices, icws, pp. 1224-1225, *IEEE International Conference on Web Services* (ICWS 2007), 2007

[6] Sanchez-Nielsen, E., Martin-Ruiz, S., and Rodriguez-Pedrianes, J. An open and dynamical service oriented architecture for supporting mobile services. *In Proceedings of the 6th international Conference on Web Engineering*, ICWE '06. ACM Press, New York, NY

[7] JSR 172: J2ME^{TM}Web Services Specification.
URL: http://www.jcp.org/en/jsr/detail?id=172 [online], *cit. 31.8.2007*

[8] kSoap. URL: http://ksoap2.seigel.info/wiki/index.php?title=Main_Page [online], *cit. 24.8.2007*

[9] BlackBerry MDS Studio: Fundamentals Guide, Research In Motion , PDF dokument