

# REMOTE ACCESS TO INFORMATION SYSTEM VIA PDA

Tomáš TUREČEK, Doctoral Degree Programme (1)  
Department of Computer Science, FEI VŠB - TU Ostrava  
E-mail: tomas.turecek@vsb.cz

Supervised by: Dr. Miroslav Beneš

## ABSTRACT

The article describes modern demands to cooperation between information systems and mobile devices. It ruminates problems with coding a distributed application which can be launched on mobile devices like PDA. The article contains information about Web services which cooperate with other applications and about few technologies that form a base of these services. An example of a practical application in the area of university information systems is shown.

## 1 INTRODUCTION

Schools and universities always needed some kind of system for maintaining information about students, their results, and many other administrative data. With growing demands to data processing and with information technology expansion, electronical information systems (IS) become university standards.

Most universities have their own IS where students can register for their exams or find out information about school and studies. All these IS are identical in one point — they almost always have some kind of web interface which is accessible through computer and its web browser only.

But computers become smaller recently. Many people own laptops or PDAs and almost everyone has a cell phone. The next logical step is to extend IS to reach mentioned mobile devices.

## 2 SITUATION ON UNIVERSITIES

A common school IS (as mentioned above) communicates with the user via the web interface using web browsers to display data. Most universities in our country have problems with insufficient number of computers publicly available to students. Classrooms are overcrowded and students have less access to these computers out of lessons. Despite the fact that the teachers are equipped well with computers, they are also often in situations,

where a computer is not effectively available and where access to the faculty IS is needed, like from lecture rooms or at home.

There exists another possibility how resources of IS could be accessed. Increasing number of students and professors have their own mobile computers — PDAs and laptops. It is only necessary to provide sufficient communication facilities.

One of possible approaches that allows to combine on-line and off-line access to an IS is based on a specialized client that is able to synchronize data over the network. For example, a teacher can download the list of students enrolled in the exam together with their current results, then update the results off-line and finally synchronize the new results with the IS.

### **3 DESCRIBING THE PROBLEM**

Every operation we do when accessing an IS from a PDA starts with request sending. The request is handled and analyzed by the server which makes appropriate operations on the database and sends back the response to the client. The response is handled and displayed by the user's PDA.

There is a problem of two remote applications which must communicate together. One of currently available solution is to use Web services.

#### **3.1 WEB SERVICES**

A *Web service* is a software system identified by URI (Universal Resource Identifier) whose public interface and bindings are defined and described by the XML. Its definition can be discovered by other systems.

The basic architecture models interaction between three roles: a *Service provider* which offers a Web service, a *Service requester* which wants to use a Web service, and a *Discover agency* which helps to find an appropriate Web service to the Service requester.

A Web service is a server application working somewhere on the Internet and listening on some port for remote invocation of its method. This method provides a service like getting current stock information, looking up in a dictionary, enrolling to the exam, etc.

When developer creates a client application, he must know how to invoke a method of the remote object. In order to define which methods a particular Web service offers and what are their interfaces, the WSDL (Web Service Definition Language) has been created as one of the XML-family languages. It describes in detail the remote service.

In case of distributed object applications it costs a lots of work to accomplish communication between remote objects. Therefore some supporting tools that facilitate the communication have been developed. One of these tools is JAX-RPC.

#### **3.2 SOAP**

The Simple Object Access Protocol (SOAP) uses XML messages to invoke remote methods.

SOAP defines uniform way of passing XML encoded data. It can be used for exchanging information in distributed environment.

SOAP is fundamentally a stateless, one-way message exchange paradigm, but applications can create more complex interaction patterns.

### **3.3 JAX-RPC**

JAX-RPC is the Java API for XML-based Remote Procedure Call which enables Java technology developers to build SOAP-based Web services. It provides the core API for developing and deploying Web services on the Java platform.

A web service endpoint a Servlet or EJB component and is described using a WSDL document. JAX-RPC requires SOAP and WSDL standards to achieve cross-platform interoperability.

In most cases, developers are not exposed to the complexity of the underlying runtime mechanisms. A JAX-RPC runtime system abstracts these runtime mechanisms for the Web services programming model, what simplifies Web service development.

## **4 PROPOSED SOLUTION**

In the previous section a few technologies has been described which helps to get data from IS offering Web services into another application. This application can be designed for and launched on PDA.

The current research at the Department of Computer Science VŠB-TU Ostrava is devoted to process modelling and implementation in the mobile environment. In cooperation with developers a project of a new generation of the faculty IS has started. This new IS will offer besides many other services a set of Web services oriented to interoperability with mobile applications running on PDAs and laptops. This section outlines how these services are implemented and which problems should be overcome during the development.

### **4.1 SERVER**

Currently running server for the faculty IS is based on the Apache/PHP/MySQL environment. As requirements on the system grow in time, the PHP language does not seem to be an ideal solution. So the next generation of the system that is under development now is built on the Java platform.

Currently implemented parts of the Java-based server provide access to resources in the IS that are published as public or internal web pages. Source data is still updated using the original PHP-based system.

The server is built around the general Model-View-Controller pattern. The model part is represented by Data Access Objects (DAOs) that encapsulate possible data sources as Java Bean components. These components offer specific properties that serve as parameters to the data source and provide a method returning a DOM representation of retrieved data.

The view part is driven by an XSLT template that is selected according to the client context. The templates are able to transform an object into one of the output formats — HTML, WML, PDF, RTF, etc.

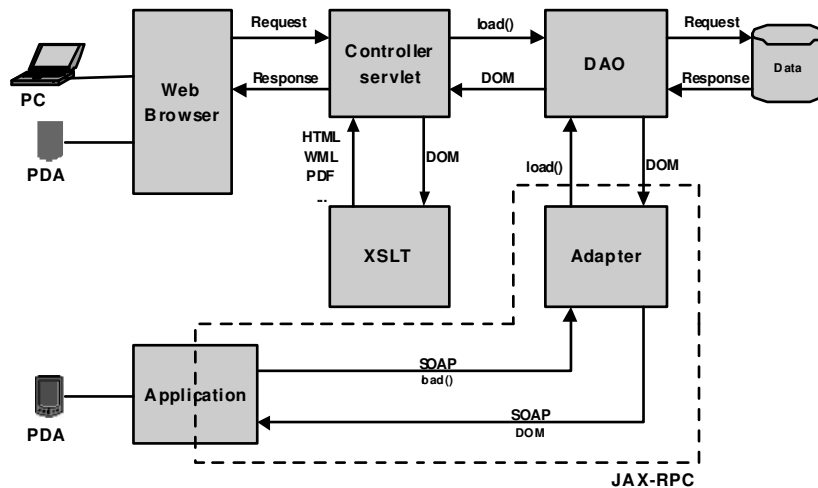


Figure 1: Server architecture

Finally, the central part of the server is a Controller servlet that processes each request in the following steps:

1. after receiving request it looks up the required DAO and an appropriate template;
2. makes an instance of the DAO class;
3. sets DAO properties according to the parameters from the user request;
4. calls the `load()` method to get a DOM representation of the DAO;
5. transforms the DOM object using the XSLT template;
6. sends the result to the client.

In case of accessing IS from PDA application, the JAXRPC model is used. The application calls method `load()` of the remote object `Adapter`. This adapter calls a DAO's method `load()` and gets back a DOM object. Then the DOM is passed to application as return value of its calling.

## 4.2 CLIENT

Clients of the faculty IS are mostly web browsers. However, to provide also an off-line access to the data stored in the system, it is necessary to develop also a thick client that is able to synchronize with the server.

In general, these type of applications are built on web services. In case the client wants to communicate with the server, it asks the user to provide user name and password, and then uses a set of services provided by the faculty IS.

As an example application built in the C# language [2] see Fig. 2. It is an application used by student to screen student's information. First screen is login screen. User writes there login and password, which is used in communication with server. Second screen shows the student's table of lectures. If there is a connection to server - synchronization with server's data can be performed.

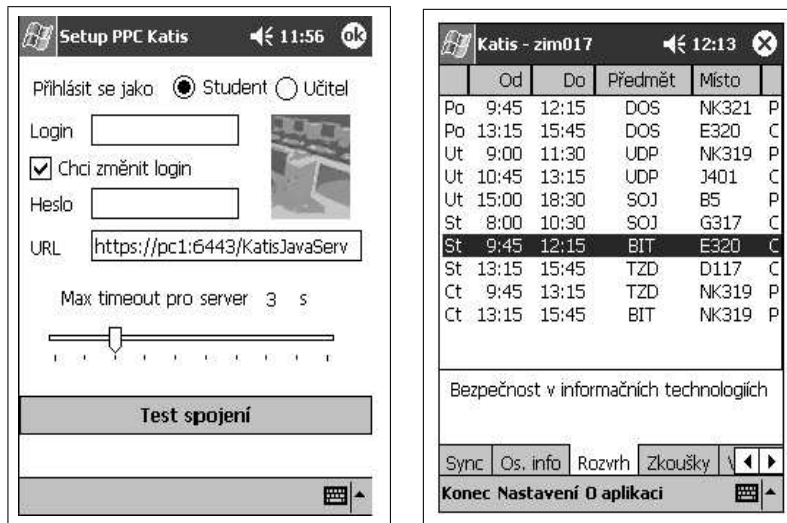


Figure 2: Client screens

## 5 CONCLUSIONS

In the article we presented our current work in the area of mobile information systems. The area of university information systems brings lot of requirements to support mobile devices, as these devices become spread among both teachers and students. Communication costs still cannot be neglected, so there is a wide area open to off-line processing combined with synchronization using web services.

This article reflects the work in progress, results of which will be publicly available to students and teachers for the next academic year. Together with the development of web services for client access to the faculty IS, the current system moves from PHP-based environment to the full Java-based solution.

## REFERENCES

- [1] Wigley, A., Roxburgh, P.: *Building .NET Applications for Mobile Devices*. Microsoft Press, Redmond, 2002, ISBN 0-7356-1532-2.
- [2] Drayton, P., Albahari, B., Neward, T.: *C# in a Nutshell*. O'Reilly & Associates, Inc., Sebastopol, 2002, ISBN 0-596-00181-9.
- [3] Farley, J., Crawford, W., Flanagan, D.: *Java Enterprise in a Nutshell*. O'Reilly & Associates, Inc., Sebastopol, 2002, ISBN 0-596-00152-5.