Mobile access, trends and technologies in modern information systems

KOZEL TOMAS, MALY FILIP, SLABY ANTONIN
Department of Informatics and Quantitative Methods
Faculty of Informatics and Management
University of Hradec Kralove
Rokitanskeho 62, 500 03 Hradec Kralove
CZECH REPUBLIC
{tomas.kozel, filip.maly, antonin.slaby}@uhk.cz http://www.uhk.cz

Abstract: Mobile technology is a buzzword of present times and mobile devices are used by almost everybody in advanced countries. But at the same time most people only make use of a tiny amount of capabilities they can offer. This contribution focuses on the most current possibilities and potential of mobile devices. We will concentrate on the possibilities and tools applicable in the process of creating mobile software applications and ways of integration of mobile clients into information systems.

Key-Words: Mobile technology, mobile device, mobile approach, web service, Java ME, Symbian, Windows Mobile, BlackBerry.

1 Introduction

Rapid development in the area of information and communication technologies has resulted in advances in mobile devices (called smart phones) with their own operating systems and rich collections of functions. These combine the potentialities of former pocket computers (PDA) and those of mobile phones. Many smart phone users do not realize the wide capabilities and possibilities of these devices. The latest worldwide statistics concerning mobile phones and their popularity [1] show a substantial decrease in the popularity of the classical PDA without a built-in telephone module. The sales of PDAs fell by 42 per cent to only 3 million pieces last year. An opposite trend is seen in smart phones. Over 118 million smart phones were sold which means an increase in sales by 60 per cent during last year alone. Smart phones have become the most rapidly growing segment of the mobile phone market, in spite of the fact that smart phones represent only ten per cent (one tenth) of all of phones sold. Most smart phones were sold in the Asia-Pacific region followed by the European market. North America succeeded in selling 10 million more clever mobile devices in the year 2007 in comparison with the previous year.

2 Areas of mobile device use

2.1 Smart phone market

The leading company in the smart phone market is Nokia, which sold 60.5 million devices last year with a market share of 52.9 per cent. Canadian Company Research in Motion (RIM) saw an upturn in the sale of their own BlackBerry communicators by 121.2 per cent last year [1], which means 4,046,860 devices sold and an 11.4 per cent market share. The third most successful company was Apple, which sold 2,320,840 iPhone devices representing a 6.5 per cent market share, together with Motorola which succeeded in increasing sales by 57.3 per cent in the period of one year. All other producers represent the remaining 22.7 per cent market share, with 8,050,920 devices sold and an increase of 28.6 per cent during last year.

As for operating systems, the most popular is Symbian, used in 65per cent of devices, followed by Microsoft with its Windows Mobile, used in 12 per cent of devices. The third most frequently used is RIM's BlackBerry system running in 11 per cent of devices followed by Apple with iPhone having 7 per cent and Linux with its 5 per cent of devices. Linux is the favourite operating system mainly in Asia, but its share in the rest of the world is tiny and can be ignored so far, but the market share of Linux based operating systems may increase after some new Linux based operating systems are introduced. An example is Access Linux Platform (APL) which is produced by the developers of former PalmSource, Linux Palm OS, or Android from Google.

Some interesting facts about Apple products should be mentioned. Apple reached third place in the world market [2], in spite of the fact that the iPhone is so far only sold in a few markets and can only be obtained from several contract operators. At the end of 2007 Apple took 28 per cent of North

American market where it holds the second place after RIM with its 41 per cent of market share. Apple has overtaken other producers using Windows Mobile as the operating system for their devices. Apple is less successful in the European market where it holds fifth place after Nokia, RIM, HTC and Motorola. iPhone is still not sold in Asia.

The Czech market offers a sufficiently rich selection of smart mobile devices. There are three mobile service providers at present: O2, T-Mobile and Vodafone. All of them offer Nokia devices, mainly N95, N73, E90, E65 and E51 with the Symbian operating system. O2 and T-Mobile providers also offer RIM BlackBerry devices and services (8800, 8310, 8300, 8100), the third provider Vodafone is preparing the same offer. It is also possible to buy devices of HTC (S710, Touch Cruise, TyTN II) with the MS Windows Mobile 6.0 operating system

2.2 Current use of mobile devices

Though there is a wide selection of mobile smart devices and rich potential for their use, the present state of their use is restricted to basic preinstalled applications of the mobile platform. Here is a list of the most commonly used capabilities of mobile devices:

- **basic telecommunication functions** – including mainly voice services, text oriented SMS and multimedia oriented (MMS) that are also accessible using common mobile phones and services offered by telecommunication providers,
- **email client** – enabling access through data services of provider, to (POP3, IMAP4) protocols and synchronization of accounts in wired mode with the aim of subsequent off-line use,
- **organizing tools** – access to diary and contacts. There is again possibility of a wired mode, but off-line synchronization with server accounts is typically used ,
- **elaborating documents** – most commonly restricted to reading documents and only exceptionally allowing modification of them,
- **web client** – typically used for accessing news portals and rarely for the access to business portals and systems,
- **entertainment functions** – which include playing games, running audio and video files, taking pictures and recording video clips with the possibility of sending them through on-line email service,
- **navigation functions** – in the case of better equipped devices with GPS.

At present even the above mentioned functions are mainly used by enthusiasts of modern technologies. We plan a wider investigation focused on discovering the real state of mobile devices use in the Czech Republic and other selected European countries.

2.3 Possibilities of advanced use

Mobile devices are much more efficient and useful than is supposed or apprehended by many users. They can offer wide possibilities of access to business information systems although they have rather restricted display and representation capability. Principle areas include:

- **online access to selected modules of information systems**– this has wide use in the area of CRM, remote monitoring and managing of the business processes, applications for pro B2B, B2C (ordering systems, marketing support …), online business data collection, etc.
- **mobile server** – enables the use of mobile devices for providing on line service in the field anywhere (implementation of a container of web services by the mobile device),
- **full integration of mobile device with groupware technologies** – mobile device plays the role of one out of many clients used for connection to an internal company communication server (for example this will include full integration of services mentioned in 2.2 with MS Exchange, IBM Lotus Domino, Novell GroupWise, etc.). In this way the mobile device serves as unique mobile communicator,
- **integration of navigation services with other functions of devices** –includes collection of data depending on the position, planning and supervision of routes (for example full support of goods distribution, location dependent CRM search, etc.).

Mobile devices and their advanced functions may be used in various layers of a company's organization structure. They can influence the structure and effectiveness of many business processes. These aspects will be of prime interests of our future research.

3 Tools

The area of application development for mobile devices is undergoing a rapid development as is the development of relevant hardware platforms. At the beginning almost every producer developed its own suite of development tools for their platform, based on the possibilities of the device's operating system.

Principle tools were based on the programming languages C, C++ (for platforms like Palm OS, Psion/Symbian, Windows CE, Pocket PC) and Visual Basic (Windows CE/ Pocket PC). The advent of mobile Java resulted in a diffusion of virtual machines for various platforms and enabled the development of portable software solutions among various types of mobile operating systems. Presently it is not imaginable to have any mobile phone or PDA without at least the possibility of Java virtual machine. Platforms based on Windows CE (i.e. Pocket PC and Windows Mobile) are suitable for developing applications in special edition of MS .NET Framework designated Compact Edition. Its portability and distribution to other platforms exists but is problematic. The following parts of the article will briefly introduce selected software platforms, their features and mainly their possibility and capability for use as mobile distant clients of information systems.

3.1 Java Mobile Edition

The basic tool for Java mobile applications development is Sun Java Wireless Toolkit [3], at present in version 2.5.2. This development toolkit consists of basic libraries and more or less comfortable utilities. Java libraries are substantially reduced in comparison with the standard edition. The compilation Java application can be achieved by the standard Java compiler and the main purpose of delivered utilities is to check the usability of the code in the mobile device and to add metadata necessary for running the application.

3.1.1 Development tools

Sun Netbeans IDE with module (plug-in) Mobility is one of the most widely used integrated development environments. Starting from version 5.0 it has been a very well developed and compact tool. It is accompanied by a visual screen designer and it also offers a generator for client proxies for web services. Another usable tool seems to be Eclipse IDE with Eclipse ME plug-in. This environment is less comfortable (offered services, complicated configuration) in comparison with the Netbeans environment.

3.1.2 Connection to IS

Until recently the most widely used method for connecting mobile clients to information systems was making use of TCP/IP sockets, but this was not supported by all mobile devices. This possibility was standardized with the MIDP 2.0 Java profile. A suite of older devices only supported connection through HTTP, and some specified application protocols had to be tunnelled via this protocol. A further step towards making use of some universal and reusable methods were specialized classes that implemented XML parser and a simple subset of the protocol SOAP on the mobile client side. KSOAP is one example of such a library.

Present trends allow mobile devices to connect directly to the web service via an extension library built-in directly in the mobile device. This extension to Java ME is specified by the document JSR-172 – WSA (Web Services API) [4]. A suite of newer mobile devices is equipped with this interface.

Though WSA does not offer the complete set of features described by WS-* standards, it is the easiest way to connect to information systems. The main advantage of this solution is the fact that web service used by the firm need not be in most cases adopted specially for the purpose of connecting mobile devices. Subsequently such a service and access to it strictly follow the regulations and requirements of SOA.

3.2 Windows Mobile

The main developmental trend of Microsoft's mobile operating systems platform is perceptible from the previous text. In the period of Windows CE all applications were developed as native applications for a specified processor. C compiler or an interpretation of Visual Basic language was used for mobile platforms. At present .Net Compact Framework is mainly used, which runs the code inside a virtual machine as the managed code. This model is similar to the mobile Java architecture.

3.2.1 Development tools

eMbedded Microsoft Visual C++ and eMbedded Visual Basic belonged to development environments used in the past. These environments were replaced by integrated environments MS Visual Studio 2003, 2005 and at present 2008 after the coming of .NET. It is not necessary to consider any other tools for this platform. Visual Studio allows the application to develop into both native or manager code, either for PDA or smart phones.

3.2.2 Connection to IS

The possibility to connect to remote information systems is similar to those of mobile Java. The solutions offered again include the scale of protocols and program classes from TCP/IP protocols, sockets, to web services. A Microsoft product called System Center Mobile Device Manager (former I.O.N.A.) [5] seems to be the most suitable means for effective implementation of the mobile approach in business. It enables

communication with a local business network via an encrypted channel and is partially inspired by RIM Blackberry services mentioned later in this article. It enables central administration, application distribution as well as the possibility to delegate rights for applications.

3.3 RIM BlackBerry solution

The Canadian company Research in Motion produces devices which are traditional mobile devices. These solutions are most popular in North America but recently have spread in Europe, including the Czech Republic, being offered by local telecommunication providers. The Blackberry mark denotes a whole family of services (Enterprise Server, Mobile Data Services, etc.), devices and tools and the main advantage is achieved by integrating all parts of the solution. RIM BlackBerry is the complex solution for establishing mobile communications for larger firms. This solution also uses so called push technologies that inform mobile devices about new events on the server side of the information system. The BlackBerry solution integrates services for managing e-mail, contacts and planning which is synchronized in real time with a dedicated server system (MS Exchange, IBM Lotus Domino, Novell Groupwise). The mobile device has the role of virtual terminal in the internal firm network and also makes use of its address space. A secure channel is provided as a part of special service bought from a telecommunication provider. Devices are centrally administered in a similar way as it is established by MS System Center Mobile Device Manager solution.

3.3.1 Development tools

BlackBerry Java Development Environment [6] is the basic development tool. It is based on Java ME, but it is enriched by a number of signed functions enabling full use of the mobile device's potential. So applications developed for Java ME can run on Blackberry devices and at the same time they can enjoy the advantage of secure data transfer. BlackBerry MDS Studio [7] offers another possibility for mobile software development. It is an integrated development environment based on the core of Eclipse. It offers a comfortable way to create web service clients using the philosophy of Rapid Application Development (RAD). Java script is widely used as the programming language. A promising emerging alternative to the web services may be support of database sources. Connectors to MS SQL Server and Oracle have appeared recently.

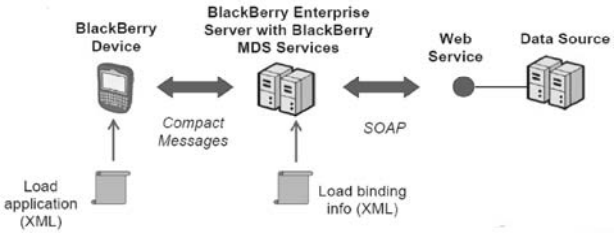

Figure 3: Architecture of BlackBerry MDS Studio [7]

BlackBerry plug-in is devoted to developers working with MS Visual Studio and enables integrated web services development with the development of mobile clients for BlackBerry.

3.3.2 Connection to IS

These two last development tools are mainly specialized for business information system access. The preparation itself consists either in the creation of access to data sources in the database or in the development or establishing access to existing web services. The solution is based on the assumption that the mobile device has the role of a terminal of the internal network and consequently it is not necessity to solve the problem of securing the transmission channel.

3.4 Symbian (UIQ)

The Symbian operating system is a proprietary operating system developed for administration of mobile devices, mainly smart phones. (Symbian is also name of the company that developed the system). The predecessor of this operating system was EPOC system for Psion computers. Users can enrich the operating system by native applications shaped for it, and these applications are dependent on the version of operating system. Symbian has its advantages and disadvantages. The main disadvantage is the problem of compatibility of applications developed for older versions of the operating system with the new ones. Another problem is the threat of virus infiltration. At present the Symbian operating system is used in Nokia, Motorola a SonyEricsson mobile phones.

S60, S80, S90 a UIQ are present versions or platforms of this operating system and a third generation of the S60 platform already exists at present. Version S80 is designed for touch operated devices and traditionally is able to cope with WiFi connection or MS Office documents. S90 is a specific version developed mainly for the Nokia 7710 communicator.

It is also possible to encounter the UIQ platform, which supports touch displays and is used with Motorola and SonyEricsson devices. UIQ can be

considered to be a graphical superstructure of the standard Symbian or S60 platform. The main difference between S60 and UIQ is in the fact that the S60 system is intended for keyboard operation and UIQ for by touch display operation.

3.4.1 Development tools

This platform offers a rich standardized application interface. It enables development of native applications in C/C++, access to all capabilities and data of the phone and access to communication interfaces (IrDA, BlueTooth, serial port). The developer can use SDK provided by producer. The platform traditionally supports Java (firstly PersonalJava 3.0, JVM and JavaPhone 1.0 API, in 2003 it was changed to pJava and JavaPhone to Java ME). Several environments for development can be employed [8], using mainly C/C++ and Java languages. The Carbide.c++ environment is specialized for C++ application development and it is based on the popular Eclipse environment. It is not recommended for large scale projects. CodeWarrior also enables the creation of native C/C++ applications. The development process is effective thanks to a number of integrated components but is often criticized as user unfriendly and unsuitable for beginners. Visual Studio .NET IDE enables creation of applications for many mobile devices including the Pocket PC. VistaMax IDE 2.0 for UIQ version 3.0 represents the best elaborated IDEs supporting rapid application (RAD) and mutual exchange with Carbide Express a CodeWarrior formats.

3.4.2 Connection to IS

The Symbian operating system can offer protocols of TCP/IP family, mainly HTTP, POP3 a SMTP for the remote connection to the information systems. Symbian architecture is comprised of the so called COMMS Infrastructure [9], which is part of the operating system including the framework for network system services and internet access. This part of the operating system includes the CommDb interface for work with communication protocols using IAP, ISP, GPRS, modem, proxy and WAP. Another important part of the operating system is called NifMan and it enables functions for basic communication with the network, for example making connections IAP, ISP and routing. The third part is a Socket Client comprising client support for EPOC sockets, TCP/IP protocol and IrDA sockets. TCP/IP is accessible via RSocket, DNS is accessible via RHostResolver interface and Generic Agent, which ensures dial-up connection to ISP are typical tools for use in the process of creating client - server applications.

4 Models of mobile client integration

In previous paragraphs we have described the main technologies, languages and tools suitable for mobile software development. In the following we introduce basic models and architectures of mobile client's implementation.

4.1 Proprietary solution of the communication

The classical solution to the problem of access to selected parts of information systems from mobile devices can be realized either by the classic socket connection and economic proprietary protocol or through the common HTTP(S) protocol. In both cases it is necessary to place the connector on the side of information system (or use a web server). The connector allows the mobile device access to the selected functions and data of information system. This solution is more suitable for very specific problems and less suitable for more common problems. In this case the client is often burdened by the quantity of application logic that it has to perform, and this fact naturally reduces performance of the device. A further disadvantage is the necessity to create a dedicated connector for connecting the mobile device only. On the other hand this solution makes the data transmission between client and information system quicker and cheaper. The model of this solution is depicted on figure 1.

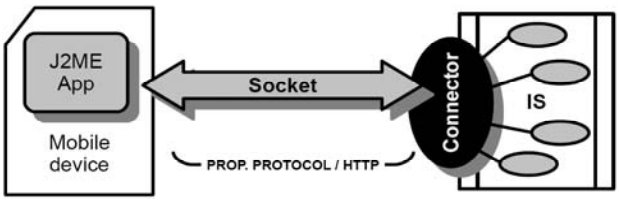

Figure 1: Proprietary solution

4.2 Making use of Web Services

Many new possibilities and opportunities for distributed applications development are created by calling remote objects through the web services (WS). Some of these address the problem of making 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 it is not necessary to implement a solution dedicated for mobile access only.
- The solution is easier to maintain and modify as the changes mainly concern the business logic of application server and so it is not necessary to distribute the changes to all mobile clients.

On the other hand it is commonly known that the SOAP protocol used for communication with the web service, which belongs to the family of XML protocols, is not a very economical method for data transmission. Two main disadvantages of this solution are:

- A large amount of data is transmitted which makes the transmission process slower and may lead to a rise in costs.
- The parsing of an XML document in the mobile device requires more computation and may slow the application.

4.3 The combination of WS and economic transmission

The combination of previous two approaches may lead to the next model. Access to the selected functions of the IS will be enabled via web services as it is described in part 2.2. Mobile clients but will not connect to these web services directly through the SOAP protocol but it will use a gateway specially designed for this purpose and it will communicate with this gate via an economic compact protocol similar to the one mentioned in 4.1. The gateway then organizes all the communication with the web services. The principle is depicted on figure 2.

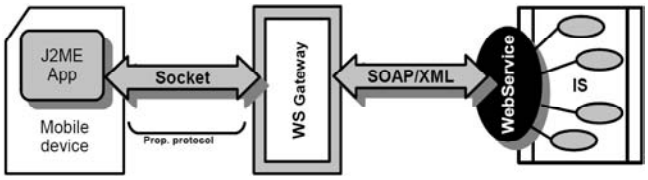

Figure 2: The gateway for connecting mobile device to web service

5 Case studies

We can demonstrate all these models of device integration on our real case studies from different areas of business.

5.1 Mobile catalogue (PalmOS)

We start with the first project implemented for the PalmOS Garnet platform. We did not mention the platform and usable tools because of the decreasing share of Palm devices on the market and because its operating system is obsolete. This case demonstrates the first model of device integration – particular protocol based on HTTP. The project grew from a mobile application developed as an off-line catalogue mirroring some functionality of a B2B on-line eShop. The need for this alternative mobile client developed because the salesmen weren't always able to access the on-line shop on the web, but they needed to collect a large number of orders throughout the day. Hand written orders on paper needed additional manual processing after the return of the person to the office. The off-line mobile client offered them to perform the following:

- View the current categorized catalogue, including prices, units, discounts etc.
- Synchronize the catalogue with the current catalogue using a connected Desktop PC
- List all the customer cards and contact details
- Prepare on site orders
- Synchronize and process all the collected orders using a desktop PC

The improving quality of mobile data services and new possibilities of connecting Palm devices to mobile networks brought the chance to enrich the application by some on-line capabilities, including:

- On-line order processing
- Verification of immediate availability of requested items
- On-line discount rates, rebate etc.

The technical solution was based on the socket communication described in subsection 4.1 using implementation of a subset of the HTTP protocol on the device side. The network connection was at first realized using a mobile phone with IrDA or Bluetooth connectivity. Later Handspring and Palm developed PalmOs based Communicators Treo which offered the ability to employ only one integrated mobile device.

Figure 3: PalmOS mobile catalogue

The architecture consists of the mobile application module developed using PalmOS Development kit in GNU C. Synchronisation of large data sets (like catalogue of items, units, customer details etc.) uses a wired cradle connection with a desktop PC through Palm Desktop Modules called Conduits. The customer can chose to process the orders either with help of the same wired synchronisation or on-line using a wireless mobile network (GSM). The overall scheme of the application architecture is shown in the figure 4.

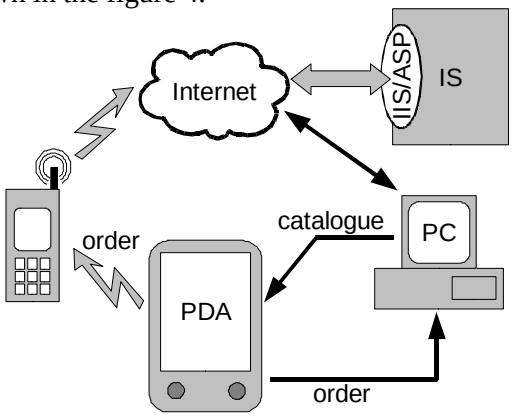

Figure 4: PalmOS catalogue architecture

5.2 Network management system (J2ME)

The main purpose of this project is to check the traffic on the local area network and to observe the state of all active and terminal network devices. The simplified architecture of this system is depicted in Figure 5. The connection to all the terminal devices is periodically tested depending on the test scheduling plan. If the connection test fails the test server starts a deep test, which includes the examination of the connection path described by the *Infrastructure database*. The first failing active network device on the path is marked as not working (or badly working - depending on some measured values) and a record is written to the *Reporting database*.

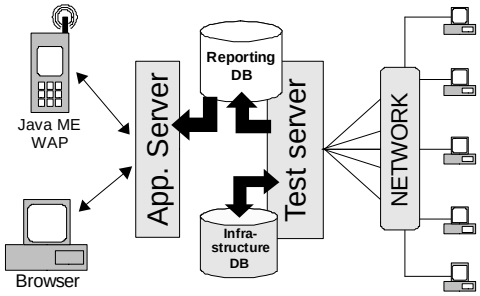

Figure 5: Network management system

The testing server continues periodic checking of that failing device, whilst the reporting server can send an e-mail notification. The reporting server offers some presentation logic which enables t WAP, J2ME and web access. The J2ME client uses a proprietary protocol model according to 4.1. The communication is secured using the HTTPS protocol and user authentication. The *Application server* is responsible for executing all the logic necessary for the clients' operation. The mobile client's functionality includes:

- the general overview of the state of all network terminal devices (PCs, printers, servers, ...),
- overview of all failing devices and
- an interface to primitive network utilities that are executed on the application server (ping etc.).

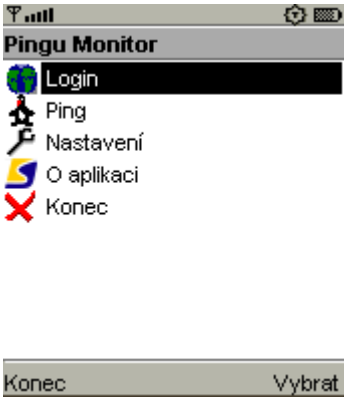

Figure 6: J2ME client of a network management system

5.3 B2B ordering system (BlackBerry)

The aim of this study was the creation of a mobile application for travelling salesmen who directly enter orders at the customer's site (*BBShop*). The concept of the application is similar to the

application mentioned in 5.1. The application offers the following functionality:

- Access to a subset of company's CRM system – i.e. contact browsing, overview of contracts, remarks; the integration with the BlackBerry communication services (allowing immediate phone calls, e-mail processing),
- Catalogue access – including simple and advanced filtering, display of customer as well as dealer prices, list of price reductions for the selected customer,
- Order processing – preparation of orders of the items selected from the catalogue, or entered using the Bluetooth barcode reader, dispatching the order and its immediate confirmation (incl. the directly available amount of items, reservations etc.).

All these functions are available via the BlackBerry device connected to the company's intranet using BlackBerry Enterprise Server (BES). The solution is based on the extensive use of the BlackBerry Mobile Data Services (MDS) module mentioned earlier. All the operations accessible through the Blackberry terminal/client are implemented in the middleware and they are exposed to the device using the web services endpoint. Although all these web services are generally re-usable and accessible from different sources, their implementation is stigmatized by the fact that all the data will be retrieved and/or processed by the mobile device. The limitations of the mobile platform in this case included:

- The recommended and default size limit for the data transfer message between BB device and BES is 32 kilobytes.
- The amount of the available internal memory of the device is limited.
- Browsing and searching large datasets on the device is slow and inefficient.
- The size of the catalogue and of other data components of the application is so large that the online synchronisation or transfer is still practically unusable in current mobile networks.

All these aspects led to the decision to implement some specific concepts of work with large datasets. It involves for instance:

- implementation of catalogue and customer list output pagination with only the current page held in the device's memory,
- simple and advanced data filtering including the filters based on the categorization of items, full text filters, barcode filters etc.
- the possibility to define and use a smaller and the user's own sub-catalogue of "favourite" items.

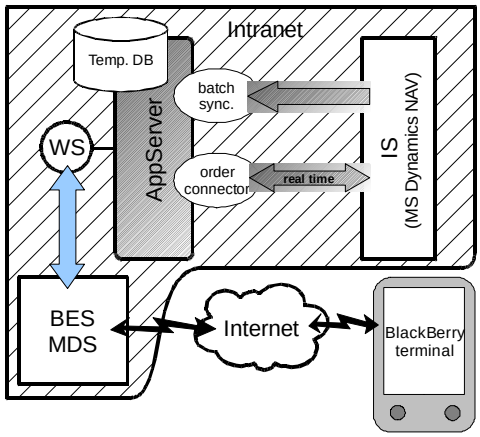

Figure 7: BBShop – BlackBerry ordering system

The overall architecture model is shown in the figure 7. It is significantly influenced by currently used Blackberry solutions. The client and communication infrastructure is implemented in the BlackBerry MDS Studio mentioned earlier. The client code is minimalistic and is mainly generated by tool wizards. The main task of the client is to collect the user's demands, to check entered data and pass them to the remote service using its generated stub. The BlackBerry Enterprise Server together with Mobile Data Services (*BES/MDS)* translates the client's request to the XML/RPC call against the Web Service Endpoint (*WS Endpoint*). The Web service delegates the responsibility to the middleware application server (*AppServer*). In case of an order request the data are pre-processed and sent through the online connector (Order connector) to the Enterprise Information System (IS – here Microsoft's Dynamics NAV). The response of the information system is then sent back to the client the same way, which includes the necessary reverse protocol translation. In other cases (non-order requests) the request is processed entirely by the application server and the data is sent back to client without any on-line cooperation with the information system. The data for the basic tables of keys are cached in a temporary database on the application server and are synchronized once a day during the night hours. This solution significantly reduces the need for direct communication with the information system.

Middleware application server was designed as a modular system allowing further extensions of functionality in the future, therefore the Spring Framework was selected for its implementation. The concept of loosely coupled software components of

Spring exactly meets the strong modularity requirement. The caching database is accessed using the Hibernate framework, which allows use of different relational database systems as back end storage engines. This solution has paid off after it was decided to change the database server technology from MySQL to Microsoft SQL Server. The need to refactor the code was minimal and the system was switched to the new database system in several minutes. Communication with the enterprise information system is based on an XML family protocol and the transmission of data is implemented using the socket connection. The entity beans of the application server are directly mapped on the XML entities with the help of JiBX framework integrated into Spring context. Web services are exposed using Axis and by sub-classing of the Spring's class ServletEndpointSupport. All the communication between BlackBerry Enterprise Server and the application server goes through the local network, so the web tier running on Apache Tomcat does not need to use very strict security policies. Further communication towards and from the BlackBerry client runs through the secured channel provided by the BlackBerry services and selected telecommunications operator.

BlackBerry also helped solve the problem of increased load on the information system in the late afternoon and early evening hours. During this time all the salesmen used to dispatch their orders which had been collected during the day. BBshop project enabled the distribution of the application load on the information system more proportionally during the day and at the same time achieved very quick distribution of the orders to the customers. On the other side the BlackBerry based solution is substantially more expensive than solutions based on other technologies mentioned and is mainly suitable for larger companies that use or intend to use BlackBerry as their basic communicator. One problem we encountered at the beginning of the projects and that we still feel as relatively serious is the missing support of connectivity detection on the device's side. BlackBerry devices store all the outgoing messages into a message queue without any regards to connectivity state. If the device is under coverage, the messages are directly sent to BES/MDS server. In the opposite case they are blocked until the connection is restored. This is convenient in many applications, but sometimes the user wants to get either immediate confirmation of the transaction or at least to know that the confirmation will be delayed. Current versions of the BlackBerry MDS Studio do not allow this function.

Figure 6: B2B Ordering system

5 Conclusion

A mobile device that uses its own operating system and enables connection to data services of providers has already sufficient equipment for the implementation of the remote access to selected parts of information systems. At present all platforms offer at least mobile Java with MIDP 2.0 profile, often with a number of useful extensions. In addition, producers of mobile operating systems usually offer development kits (SDK) enabling application development in native code of the platform or offer their own virtual machines which enable the running of operating systems and applications on various platforms. Web services and accompanying technologies offer a large potential for access to them from mobile devices. These technologies do not usually implement complete collection of WS extensions but are sufficient in most cases.

The essential problem of mobile access to business data is the safety factor. This problem includes security of the transmission channel, security of data in a connected or disconnected mobile device and last but not least central administration of user settings, rights and methods of accessing mobile applications. Wider use of mobile devices is unthinkable without implementation of a suitable safety policy and regulations. RIM has been offering comprehensible solution based on BlackBerry services for a long time. Microsoft is starting to offer an integrated solution thanks to its product System Center Mobile Device Manager. These solutions are not and in near future will not be the cheapest ones and so wider implementation of such technologies is mainly expected for medium size and larger companies.

Our main interests in the future research will cover following items:

- The ways and possibilities of mutual influence and synergy of the business management area and area of mobile technologies
- Basic models of interaction between mobile device and business information systems so that there business processes of firms mainly processes which include elements of real mobility could be supported effectively
- Complex searching and mapping of real business processes and finding the groups of processes which still don't have support and alternative in existing mobile information systems and analysis of conditions which either support or prevent mobility of processes
- Design of basic architectures and models of information systems, which will follow detailed analysis of the processes and will be based on massive, meaningful and reasonable use of mobile attitudes.
- Mobility and elements of Artificial Intelligence – mobile software and mobile devices, mobile agents, contribution to Ambient Intelligence.
- Synthesis of previous results coming from various areas of mobility research

Thanks:

This article is supported by the grant no. 402/08/1046 „Models of firms with mobile oriented architecture" of the Czech Science Foundation.

References:

[1] Novak, A: Nokia a Symbian stale vladnou smartphonum. *Palmare cz* URL:http://palmare.idnes.cz/nokia-a-symbian-stale-vladnou-smartphonum-apple-v-usa-predstihl-microsoft-1tn-/tech-a-trendy.asp?c=A080206_151031_tech-a-trendy_ada, cit. 2.3.2008.

[2] Skrabalek, J. Chytre telefony: iPhone vladne vsem. URL:http://www.mobilmania.cz/ default.aspx?article=1117736, *Mobilmania cz*, cit. 2.3.2008.

[3] Rančić, D., Predić, C., Mihajlović V. Online and Post-processing of AVL Data in Public Bus Transportation System, *WSEAS Transactions on Information Science & Applications*, Issue 3, Volume 5, March 2008, pp. 229-236, ISSN: 1790-0832229

[4] Soo-Hwan, K., HyeonCheol, K., Jin-Tae, L., Seon-Kwan, H., Chul-Hyun, L. A Design of u-Learning System based on Memory Theories in Ubiquitous Environments, WSEAS Transactions on Advances in Engineering Education, Issue 2, Volume 5, February 2008, pp. 77-82, ISSN: 1790-1979

[5] Sun Java™ Wireless Toolkit for CLDC. URL: http://java.sun.com/j2me/docs/sun_java_wireless_tlkit.pdf [online], cit. 2.3.2008

[6] JSR 172: J2ME™Web Services Specification. URL: http://www.jcp.org/en/jsr/detail? id=172 [online], cit. 31.1.2008

[7] Microsoft System Center Mobile Device Manager, URL:http://www.microsoft.com/cze/ windowsmobile/mobiledevicemanager/

[8] BlackBerry Java Development Environment Fundamentals Guide, PDF, http://www.blackberry.com

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

[10] UIQ Developer Community: Integrated Development Environments(IDE). URL: http://developer.uiq.com/devtools_tools.html, cit. 2.3.2008.

[11] Comms Infrastructure in C++ API guide. URL: http://www.symbian.com/developer/techlib/v70sdocs/doc_source/devguides/cpp/CommsInfrastructure/index.html, cit. 4.3.2008.

[12] Zaigham, M.: Software Products and Technologies for the Development and Implementation of SOA, *WSEAS Transactions on Computer Research*, Issue 1, Volume 3, January 2008, ISSN: 1991-8755