INFORMATION SYSTEMS SUPPORT ON MOBILE DEVICE PLATFORM

Java SCADA Client/Server model and .NET localization enhancement

Ondřej Krejcar, Jindřich Černohorský

VSB Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, Department of Measurement and Control, Centre for Applied Cybernetics, 17.listopadu 15, 708 33 Ostrava-Poruba, CZECH REPUBLIC

Keywords: Mobile device, PDA, WM2003, Linux, Java, J2ME, .NET CF, OpenNetCF, Location-aware services.

Abstract: The paper deals with programming possibilities of mobile devices. It discusses the relationships with control systems and problems with solutions of possible situations arising from their design or their operation. It is focused mainly on Java language and use of created SCADA based application on wide scale of mobile devices without any changes of source code. The proliferation of mobile computing devices and local-area wireless networks has fostered a growing interest in location-aware systems and services. Another area of interest is in model of radio-frequency (RF) based system enhancement for locating and tracking users of our information system inside buildings. User location is used to data pre-buffering and pushing information from server to user's PDA.

1 INTRODUCTION

Programming of personal digital assistant (PDA) is possible in two ways. The first and the best known way is based on MS Windows CE, now renamed to Windows Mobile for Pocket PC. Contemporary version of Windows Mobile is 2003 SE (WM2003). The second way is based on Linux distribution. OS Win CE is used not only in pocket computers, but it can be found it in other devices. There are two usable and supported programming platforms. The first one, .NET framework, is otherwise younger, but more preferred by Microsoft. It can be installed on Win CE on PDA. The .NET platform hasn't however portation to Linux. The second way is based on Java language. Java can run on both OS. Program written in Java is possible to start on every processor on which virtual machine of Java called Java Virtual Machine (JVM) is running. JVM is virtual (software or hardware) processor on which is possible to start the program or applet written in Java (java applet). JVM is implemented in a number of embedded devices, servers, mobile phones and PDA's. Next benefits are in uniform language platform for development of company systems and a reuse of code.

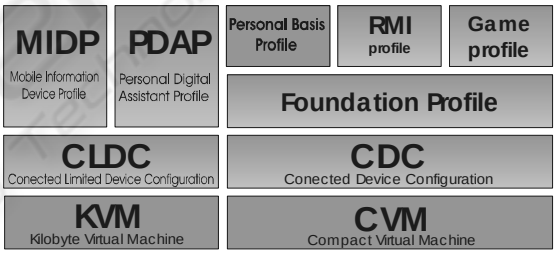


Figure 1: Basic J2ME structure

2 JAVA 2 MICRO EDITION - J2ME

Java language is currently distributed in three different packages various in their sets. One of them is known as Java microedition platform. This distribution started up with the aim to consolidate various offshoots of Java language for small devices, which aren't contained in the standard edition. This establishment encompasses devices with various properties. Microedition isn't one specification like standard edition, but it is a set of various configurations and profiles. These configurations designate basic set of libraries and properties of devices, which refine profile.

Krejcar O. and Černohorský J. (2005).
INFORMATION SYSTEMS SUPPORT ON MOBILE DEVICE PLATFORM - Java SCADA Client/Server model and .NET localization enhancement.
In *Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics - Signal Processing, Systems Modeling and Control*, pages 205-210
DOI: 10.5220/0001179202050210

2.1 Mobile Information Device Profile

MIDP profile specifies Connection Limited Device Configuration (CLDC) for use in smallest devices, like usual mobile phones. Just this platform pleases largest focus, because it is concerned in a body of extensive devices. The MIDP profile adds to hardware specifications a minimum requirement on display size (96 x 54 pixels) and the possibility to control device by character keyboard or by screen touching. It also requires at least 8 kB stable memory for saving data application. In version MIDP 2.0, whose specification is already available, important functionality is improved, like interface by sockets or sound control. Practically each of mobile phones belongs to MIDP category. Application for this category is called midlet according to basic class of MIDP profile.

3 JAVA AND LINUX ON PDA IPAQ

The most common variant is using Linux on classical Pocket PC. Since Pocket PC's are in most cases on purchase already equipped with Windows CE system, it is necessary to replace it with desired Linux distribution. This is user's exacting operation requiring higher knowledge level in the area of installation and system repair. There are now multiple projects focussed on development of Linux implementations for the iPAQ, including one from the Hewlett-Packard sponsoring handhelds.org site. In particular, this Familiar Project provides comprehensive free and open support for the iPAQ.

3.1 HP iPAQ platform

HP iPAQ 5450
The iPAQ 5450 model has a 240 x 320 pixel backlit colour LCD screen and is powered by a 400MHz Intel Strong-Arm processor with 64MB of RAM and 64MB of flash memory. This device has integrated support for wireless solutions as Wi-Fi, Bluetooth and infra red.

On this model of iPAQ we have installed a Familiar Linux distribution on which is running JVM needed for SCADA application. We use HP iPAQ H4150 on which is running WM2003 operating system to tests created applications.

HP iPAQ 4150
HP iPAQ 4150 is classical PDA from Hewlett-Packard too. It has the same parameters as 5450 model. Only flash memory (NAND flash) is smaller in compare with.

This system is equipped by WM2003, and a special virtual machine J9 from IBM Company.

3.2 WebSphere Everyplace - J9 VM

If we want to work with an application written in Java, we have to install Java runtime Environment (JRE). JRE includes JVM, which is typed and compiled for existent platform, respective concrete processor of given Pocket PC, how it was already described above.

The J9 VM is the core of WebSphere Everyplace Micro Environment (WEME), the IBM implementation of the JVM Specification, Version 1.3.

The J9 VM and Java Class Libraries (JCL) comprise the J9 runtime environment. The J9 runtime environment is J2ME compliant and contains CLDC and CDC based technologies. In addition, the WebSphere Everyplace Custom Environment (WECE) is a combination of the J9 VM and IBM custom libraries.
The WEME product is supported on a variety of:

- Operating systems (including Microsoft Windows, Linux, PalmOS, OSE, Rex, VxWorks, PocketPC, Symbian, QNX and Nucleus)
- Hardware architectures (including Intel x86, xScale. ARM, MIPS, SH4, and PowerPC)

On J9 VM we can run application created for mobile phone under MIDP specification without any change of source code. Developer must create only one product in one specification and it is runnable on PDA, mobile phone or on other embedded device with running J9 VM.
Space required to install J9 VM is:

- J2ME Mobile Information Device Next Generation (JSR-118) MIDP 2.0 – 3MB
- J2ME Personal Profile 1.0 (JSR-62) – 8MB

4 SCADA

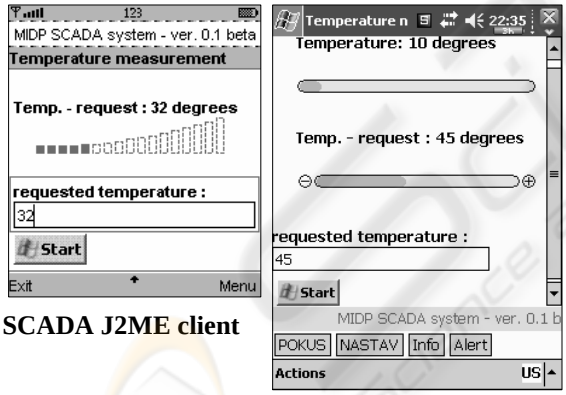
SCADA stands for Supervisory Control And Data Acquisition. As the name indicates, it is not a full control system, but it rather focuses on the supervisory level. As such, it is a purely software package that is positioned on top of hardware to

which it is interfaced, in general via Programmable Logic Controllers (PLCs), or other commercial hardware modules.

4.1 SCADA model

We have created new SCADA client – server application based on SCADA systems principals. This model is very useful and scalable for many tests of embedded devices (SCADA clients). Model of SCADA server's doesn't contain some standards of SCADA based systems like real-time database and historical trends. We plan to extend our SCADA application by these features in the future.

SCADA J2SE server
Model of SCADA server was created using Java application base in standard IDE environment of Borland JBuilder X. Application is based on J2SE functionality and library. Some options of user interface like graph are developed for better visualization of measured values. The needed values like temperature are generated by special created algorithms, or are set by user. User can set value by adjustable scrollbars, and their value is visualized online by other graphical element.



SCADA J2ME client

Figure 2: J2ME mobile phone & PDA client

Model of SCADA client was developed in the same IDE JBuilder X as server above. As development platform the Java 2 Mobile Environment specification (J2ME) was used. This specification allows running applications on mobile phone and on PDA with installed JVM. In our case it is IBM J9 VM. This is very important for code reuse and safety of our application. Application is created by high level MIDP components interface and other MIDP component usable for SCADA systems. This guaranties trouble free portability between mobile devices.

High level components of SCADA system
The first component which is used in SCADA client is text field. This can be used to setting request for sending value from SCADA server. Gauge is the second component and it is used for visualization of actual temperature value. The gauge is in enabled state, what means, that we can change the value of this component. Example of these two components is shown on figure 2 (mobile and PDA).

Other high level component is the image item component, which is useful for graphical representation of some problem. Notice that there is a different presentation of application between mobile phone and PDA. In mobile phone the buttons are on tool list, meanwhile in PDA they are visible as self button.

Alarms in J2ME SCADA applications
Alarms are well known and used in SCADA systems. J2ME platform has this support too. Alarms are in Alert class, which is subclass of Screen class. Alert's behaviour is the same as known dialogs, but it has reduced functionality. There are many types of alerts:

- ALARM - alarm
- CONFIRMATION - confirmation
- ERROR - error
- INFO - information
- WARNING - notification

Alerts are characterised by many parameters like their name, message for user, image and timeout.

5 LOCALIZATION ENHANCEMENT

The proliferation of mobile computing devices and local-area wireless networks has fostered a growing interest in location-aware systems and services. A key distinguishing feature of such systems is that the application information and/or interface presented to the user is, in general, a function of his physical location. The granularity of location information needed could vary from one application to another. For example, locating a nearby printer requires fairly coarse-grained location information whereas locating a book in a library would require fine-grained information.

While much research has been focussed on development of services architectures for location-aware systems, less attention has been paid to the fundamental and challenging problem of locating and tracking mobile users, especially in in-building

environments. We focus mainly on RF wireless networks in our research. Our goal is to complement the data networking capabilities of RF wireless LANs with accurate user location and tracking capabilities for user needed data pre-buffering. This property we will use as information ground for our information system.

5.1 Data Collection

A key step in proposed research methodology is the data collection phase. We record information about the radio signal as a function of the user's location. The signal information is used to construct and validate models for signal propagation.

Among other information, the WaveLAN NIC makes available the signal strength (SS) and the signal-to-noise ratio (SNR). SS is reported in units of dBm and SNR is expressed in dB. A signal strength of s Watts is equivalent to 10*log10(s/0.001) dBm. A signal strength of s Watts and a noise power of n Watts yields an SNR of 10*log10(s/n) dB. For example, a signal strength of 1 Watt is equivalent to 30 dBm. Furthermore, if the noise power is 0.1 Watt, the SNR would be 10 dB.
The WaveLAN driver extracts the SS and the SNR information from the WaveLAN firmware each time a broadcast packet is received. It then makes the information available to user-level applications via system calls. It uses the wlconfig utility, which provides a wrapper around the calls, to extract the signal information.

6 PDPT FRAMEWORK

Predictive data push technology (PDPT) framework is name of our project. Basic idea is in connection between location information and data push technology. When we know user position and user track, the data which could be needed by the user in the future is pushed to his mobile device. This part of project is being developed under .NET CF and OpenNETCF on mobile device (client) side. .NET language is basic for server side application. It is supposed that in the future will be possible to make an interconnection to existing Java SCADA application discussed before.

6.1 Pushing in Intranets

Intranets are simply local 'internets' close inside various organisations. They allow transfer of information between users. Important company news and information can be pushed straight to the Screen of the employees. This could be done in form of screen savers, tickers (scrolling bars of text) or by some client application. A good example of a useful application of push within companies is automatic notification of events. For example, by linking a push system to a database of stocks and components in a manufacturing company a manager could be alerted if the number of a particular component drops below a certain level.

When some intelligence is added to push system, the very useful system may be created. This is what we do. Location information about users is added to server as intelligence for push technology.

6.2 Predictive data push technology

This part of project is based on model of location-aware enhancement, which will be used in created information system for adding user location informations. These information about will be useful for next generation of framework to increase real dataflow from wireless access point (server side) to PDA (client side). Primary dataflow is enlarged by data pre-buffering. These techniques form the basis for predictive data push technology (PDPT). PDPT copies data from information server to clients PDA to be on hand when user comes at desired location.

The benefit of PDPT consists in reduction of time needed to display desired information requested by user command on PDA. Time delay may be from a few seconds to number of minutes. It depends on two aspects. First aspect is the quality of wireless Wi-Fi connection used by client PDA. A theoretic speed of Wi-Fi connection is max 825 kB/s. However, the test of transfer rate from server to client's PDA, which we have carried out within our Wi-Fi infrastructure provided the result speed only 160 KB/s. Second aspect is the size of copied data.
The application (locator)

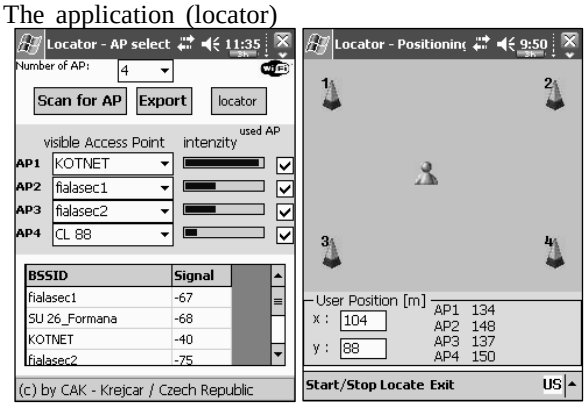


Figure 3: Locator – AP intensity & Positioning

based on .NET language is now created for testing. Current application records just one set of signal strength measurements. By this set of value the actual user position is determined.

6.3 Wireless location architecture

Another very important part of project is design of Wireless Location Architecture (WLA). WLA define structure for data store in database. Structure is defined as data levels in building plan for example.

Example of function: User location is determined and analyzed. Server activates the PDPT and pushes data to PDA. As first the data about Building Block will be copied. As next the Block Floor data and finally the data about user occurred cell will be copied.

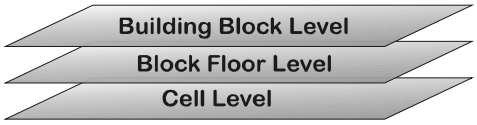


Figure 4: WLA for building

6.4 Framework design

PDPT framework design is based on most commonly used server-client architecture. To process data the server has online connection to Control System. Data from technology are continually saved to SQL Server database (WLA architecture). The part of this database (desired by user location or his demand) is replicated online to client's PDA where it is visualized on the screen. User PDA has location sensor component which continuously sends to framework kernel information about nearby AP's intensity. The kernel processes this information and makes a decision if and how a part of WLA SQL Server database will be replicated to client's SQL Server CE database.

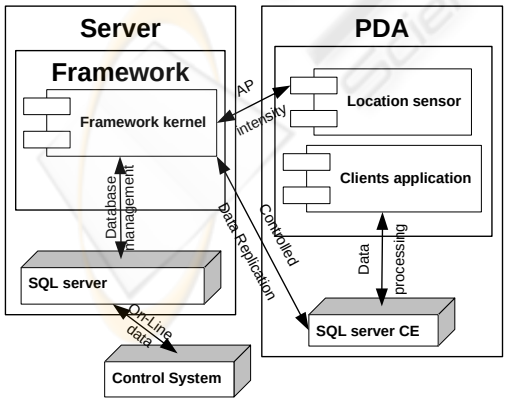


Figure 5: System architecture – UML design

These kernel decisions constitute the most important part of whole framework because the kernel must continually compute the user position and track and make a prediction of his future movement. After doing this prediction the appropriate data (part of WLA SQL Server database) are pre-buffered to client's database for future possible requirements.

7 CONCLUSION

The paper deals with programming possibilities of control systems and problems with solutions of possible situations. The main objective is to show how to use information systems on concrete mobile device (mobile phone and PDA iPAQ) in context of use Java language. The main advantage is in possibility to use created java based SCADA application on wide scale of mobile devices without any changes of source code.

Other in this paper presented feature is the enhancement of information system for locating and tracking of users inside a building. It is possible to locate and track the users with high degree of accuracy. The median resolution of the system is in the range of 2 to 3 meters, about the size of a typical office room. Our practice and experiments indicate that it is possible to build an interesting class of location-aware services, such as printing to the nearest printer, users navigating through a building, etc., on an RF wireless LAN. User location is used to data pre-buffering and pushing information from server to user's PDA. Data pre-buffering is most important technique to reduce time from user request to system response. This, we believe, is a significant contribution of our research.

ACKNOWLEDGEMENT

This work was supported by the Ministry of Education of the Czech Republic under Project 1M6840770004

REFERENCES

Krejcar, O., 2004. Mobile information system via wireless technology. PDS'04, IFAC Workshop on Programmable device and Systems. Cracow, Poland.

Yuan, M., 2003. *Enterprise J2ME: Developing Mobile Java Applications*, Prentice Hall, San Francisco, USA.

Wilding-McBridge, D., 2003. *Java Development on PDAs*, Addison-Wesley, Boston, USA

Reynolds, J., 2003. *Going Wi-Fi*, CMP Books, San Francisco, USA

Wigley, A, Roxburgh, P., 2003. *ASP.NET applications for Mobile Devices*, Microsoft Press, Redmond, USA

Tiffany, R., 2003. *SQL Server CE Database Development with the .NET Compact Framework*, Apress, USA

Thilmany, Ch., 2003. *A .Net Patterns: Architecture, Design, and Process*, Addison-Wesley Professional, USA