Design and Development of "GlobeTrotter" – An Innovative Location-Based System using J2ME

Anna Isakow and **Hao Shi**

School of Computer Science and Mathematics
Victoria University, Melbourne, Australia

Summary
GPS (Global Positioning System) is the most popular system for determination of location and navigation in the world. GPS works in all weather conditions and has become a widely used aid to navigation systems worldwide and a useful tool for map-making, land surveying, commerce and scientific uses. More recently GSP has been integrated into mobile systems such as car navigation system, which use maps directly stored on a mobile device. However, in order to catch up the progress of map development, users have to update their maps frequently and pay high charges to their providers. There is urgent need to develop a new dynamic mobile application to allow users to get access to the latest information without any additional cost. A location-based system called "GlobeTrotter" based on J2ME architecture is proposed. It integrates GPS and Internet mapping technologies using J2ME (Java 2 Micro Edition). It aims at display the current user location on the screen of a mobile system in real time on a map like Google Maps, Yahoo! Maps, MSN Maps or Ask.com Maps. In the paper, first J2ME is examined. Then current and future applications for mobile devices are summarized. The design and development of "GlobeTrotter" and integration of Internet mapping with mobile devices are presented in detail. The successful implementation of "GlobeTrotter" provides an innovative solution to current and future development of J2ME-based applications and it has an enormous economic potential to meet customers expectations in mobile services.

Key words:
J2ME, mobile applications, GPS, Internet mapping

1. Introduction

Within the last 20 years, mobile phones spread like wild fire. With more than 2 billion phones around the globe and more mobile than fixed line subscribers, mobile phone industry is the most growing industry in the world. The development progressed from unhandy, simple phones to small all-rounders with high-resolution colour display, organizer, integrated camera and GPS receiver. It is expected that this trend will proceed and mobile devices will have more memory storage and functions.

Due to the enormous performance improvement of devices and the continuously increasing mobility of users, the mobile phoning has already become an implicitness. Mobile systems which can determine the geographical location and offer new services according to this special position will become very popular.

High investments have been made in the development of the expensive broad band networks like UMTS. It raised great expectations on new applications by network carriers. Mobile solutions in areas like shopping, advertising and safety conceal powerful revenue potentials. In searching for a large user market like SMS, location-based services become one of the hottest topics.

Nowadays, almost all mobile phones available on the market support the programming language Java for Java 2 Micro Edition (J2ME). J2ME allows to implement platform independent applications for mobile devices. Many J2ME games already exist and enjoy great popularity especially among youth. Java has become the object-oriented programming language for developers to implement new mobile applications [1].

2. Overview of J2ME

At the JavaOne Conference in June 1999, Sun Microsystems announced a new edition of the Java 2 platform - the J2ME. The purpose of the Micro Edition is to enable Java systems to run on small computing devices. J2ME does not define a new programming language. It simply applies existing Java technology to handheld and embedded devices. Unlike J2SE or J2EE, J2ME is neither a piece of software, nor is it a single specification. Instead, J2ME is a platform, a collection of technologies and specifications that are designed for different parts of the small device market. Because so many different devices use J2ME, it would be impossible to create one solution fitting all devices. Therefore, it is divided into different and a steadily growing variety of configurations, profiles and optional packages as shown in Fig. 1.

Nowadays, J2ME has become an organized architecture for mobile devices. It includes sets of Java Application Programming Interfaces (APIs) for high-end PDAs and

Manuscript received March 5, 2008
Manuscript revised March 20, 2008

embedded devices and for more constrained devices such as mobile phones, low-end PDAs and headless devices, those without display or user interface facilities.

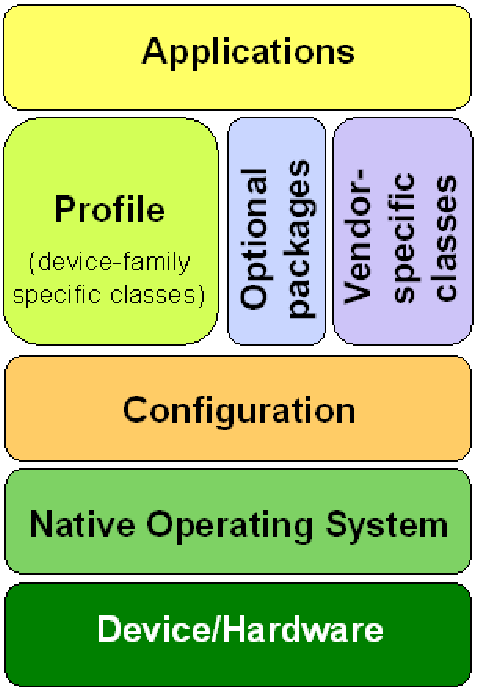

Fig. 1: J2ME software layers [1]

Configurations provide core functionality but no services for managing the application life-cycle, for driving the user interface, for maintaining and updating persistent data on the device or for secure access to information stored on a network server [2] These types of functionality are provided by profiles or optional packages.

A profile adds more specific APIs to the core set provided by the configuration. It supports specific uses of devices and provides functionality missing from the underlying configuration.

Optional packages add special services that are useful on devices of many kinds but that are not necessarily available on all of them. Usually, applications are based on a configuration appropriate to the desired category of target devices and on a profile that supports the software's basic functionality and optional packages which support needed specialized functions like messaging or multimedia [2] A profile is based on top of a configuration and adds APIs for user interface, persistent storage and other classes needed to develop running programs. One device can support multiple profiles.

In addition to J2ME's configurations, profiles and optional packages as described above, Fig. 1 shows that applications may include vendor-specific classes when required. Vendor-specific APIs are extensions to configurations and profiles. The standard J2ME does not consist of vendor-specific interfaces but they can be added and extend the functionality that's specific to a given device, for example APIs to control the radio transceivers of a device [2]

3. J2ME-based mobile applications

There are many different applications available on mobile phones, however it is difficult to estimate the present usefulness of such services. As decisions about design, price and features cannot be made without a deep understanding of consumer behaviour, an analysis of customer expectations of mobile phones should be taken into account.

The Boston Consulting Group surveyed expectations of customers in regard to mobile applications [4] Its results are presented in Fig. 2. Over 1800 respondents have been interviewed in the USA, Japan, Germany, France, Sweden and Australia in November 2000. Even if this study has been made seven years ago, it is still convincing. For today's mobile users, the desire for basic communication and information-gathering services ranks ahead of motivations that are focused on entertainment and bargain hunting. In general, current users regard mobile services as effective tools for very specific situations. The mobile device is not seen purely as an alternative to the computer. Instead, users view it as a supplementary resource they apply for different tasks.

At the present day, the majority of J2ME programs are games like puzzles, flight emulators and sport games. Some of the games are already installed on the mobile phone when customers buy them. Games which are not included in the purchase pack, can be directly downloaded from the Internet and installed on the device. Such games enjoy great popularity especially among young people. Java for mobile phones is believed to be the standard in game programming.

An important area of J2ME applications is also the mobile shopping. The mobile phone could display its owner information about special discounts at supermarkets of the customer's choice. It could also locate the nearest cheap gas station. Another useful service for countries like Germany where shops are not opened around the clock could help in finding an emergency-pharmacy opened at night hours and on the weekends.

Another conceivable use would be an emergency system for skiers. Before departure, a skier can register at a service centre. The position of the registered skier would be traced and in case of an avalanche he can be found more quickly. If a skier would try to access the avalanches zone or in case of the weather change and high avalanche danger, the service centre can contact this skier.

On the one side, the tracking of a person over GPS receiver is highly controversial because of individual legal reasons. On the other side, the tracking offers many useful applications which should be considered of importance for the future. For example, it would be technically thoroughly possible to develop a password protected online system that graphically displays the position of the partner. In the same way parents could locate the current position of the child lost in the supermarket.

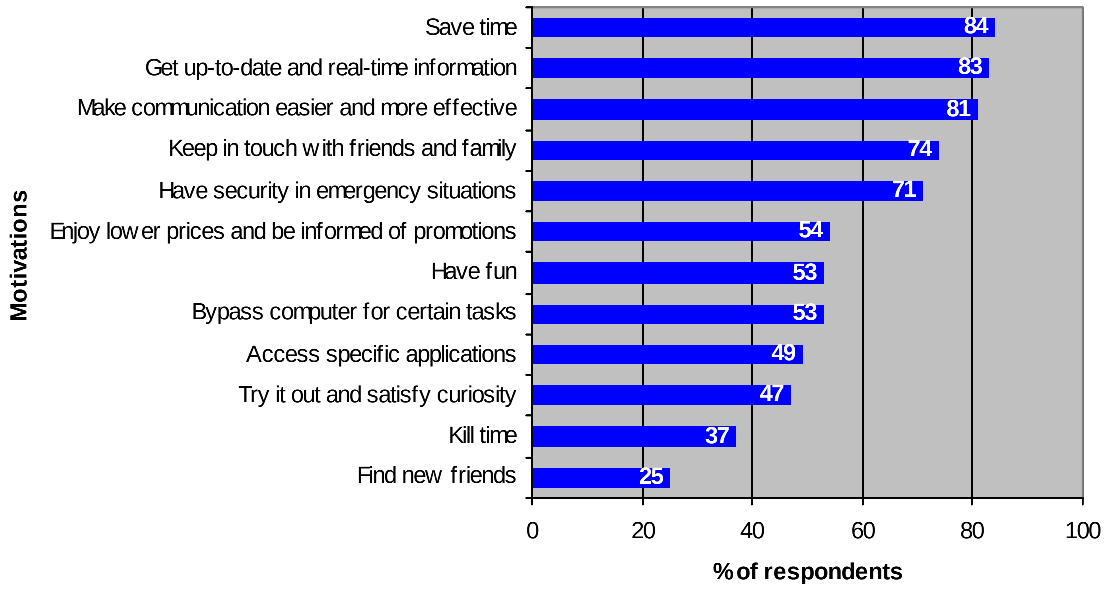

Fig. 2: Motivations for usage of mobile phones

4. Overview of GPS

The Global Positioning System (GPS) is the only fully functional Global Navigation Satellite System (GNSS). GPS is a way of locating a receiver in a three dimensional space anywhere on the Earth. Using a constellation of at least 24 medium Earth orbit satellites which transmit precise microwave signals, the system allows a GPS receiver to determine its location, speed, direction and time [5]

GPS was developed in the 1970s by the United States Department of Defence (DoD) and was officially named Navigation Satellite Timing and Ranging Global Positioning System (NAVSTAR GPS). The original usage of GPS was to identify the exact position of military units and was a great help to the United States during the Persian Gulf war in 1991 [6].

As GPS works in all weather conditions, it has become a widely used aid to navigation systems worldwide and a useful tool for map-making, land surveying, commerce and scientific uses. There are many systems like scientific study of earthquakes and synchronization of telecommunication networks using GPS for getting a precise time reference.

GPS receiver calculates its position by measuring the distance between itself and three or more GPS satellites. Measuring the time delay between transmission and reception of each GPS microwave signal gives the distance to each satellite, since the signal travels at a known speed. This information is available around the clock and is not dependent on the weather or location.

5. GPS data format

The National Marine Electronics Association (NMEA) is an organization which has defined a data specification for communication between marine electronic devices, such as echo sounder, autopilot, fish finders, GPS receivers and many other types of appliances.

GPS receiver communication is defined within the NMEA specification. Most computer programs that provide real time position information understand and expect data to be in NMEA format. NMEA 0183 consists of sentences and uses a simple American Standard Code for Information Interchange (ASCII) text to define position data which is transmitted in a sentence from one talker to one listener at a time [5] As Fig. 3 illustrates, GPS receiver delivers many sentences in different formats.

Each of the sentences consists of different elements in a fixed order like sentence identifier, time, latitude, longitude, GPS quality indicators, number of used satellites, relative accuracy of horizontal position, altitude, Geoidal height, checksum and others [5]

In order to display the user precise position on the Google Maps, "GlobeTrotter" system requires four elements of the GPGGA sentence, i.e. latitude, latitude direction, longitude and longitude direction. Latitude direction is always either South or North. Longitude direction can only be East or West.

```
$GPGGA,001932,3747.5489,S,14453.7689,E,1,3,,-74.9,M,-4.2,M,,*58
$GPRMC,001932,A,3747.5489,S,14453.7689,E,0.55,97.26,301007,,,N*
D
$GPVTG,97.26,T,,M,0.55,N,1.02,K,N*0B
$GPGGA,001933,3747.5477,S,14453.7575,E,1,3,,-90.0,M,-4.2,M,,*5B
$GPRMC,001933,A,3747.5477,S,14453.7575,E,0.68,98.32,301007,,,N*
9
$GPVTG,98.32,T,,M,0.68,N,1.27,K,N*08
$GPGGA,001934,3747.5466,S,14453.7493,E,1,3,,-100.1,M,-4.2,M,,*6C
$GPRMC,001934,A,3747.5466,S,14453.7493,E,0.72,98.15,301007,,,N*
9
$GPVTG,98.15,T,,M,0.72,N,1.33,K,N*03
$GPGGA,001935,3747.5468,S,14453.7492,E,1,3,,-101.5,M,-4.2,M,,*67
$GPGSA,A,1,,,,,,,,,,,,,,*1E
$GPGSV,3,1,09,22,74,208,18,14,49,359,,03,48,266,21,18,44,130,37*74
$GPGSV.3.2.09.19.32.231..21.29.071.40.01.18.343..09.16.106.34*70
```

Fig. 3: GPS data output

As the received location data is not in the data format required for Google Maps, the "GlobeTrotter" system converts it.

6. System configuration

Before implementation of the "GlobeTrotter" system, the installation of required software has to be successfully done. Following software is necessary:

1) Java Development Kit (JDK) in version 1.5.0_13
2) Eclipse Software Development Kit (SDK) version 3.3
3) Eclipse ME version 1.7.6 or later
4) Java Wireless Toolkit for CLDC in version 2.5.1

7. Design and development of "GlobeTrotter"

The developed "GlobeTrotter" system is based on the J2ME architecture and its components. It also combines different technologies like GPS, NMEA 0183 and Google Maps to show the current user location on a digital map on a PDA using Sun Wireless Toolkit.

The position data is automatically determined using a GPS receiver and translated into a format used by the specific map provider. The system consists of a client application "GlobeTrotter" running on the emulator and using a server library J2MEMAP on the server of http://www.8motions.com/. This system has been designed as a framework which makes it possible to extend it using additional modules and apply for other J2ME applications.

The order of execution within of the "GlobeTrotter" system is as follows:

1) GPS receiver receives position data from several satellites.
2) It calculates the exact location on the Earth.
3) GPS receiver sends NMEA 0183 sentences via Bluetooth connection to the laptop computer.
4) "GlobeTrotter" system running on the laptop computer receives the position data.
5) The system translates it to the data format acceptable by the map provider like Google Maps.
6) The "GlobeTrotter" system establishes an Internet connection.
7) It sends HTTP requests to the 8motions server providing it with the required map name, zoom, latitude and longitude.
8) 8motions server requests maps from the original server.
9) It responds to the "GlobeTrotter" system with required map tiles.
10) Emulator application embeds the received maps into the screen and centres them around the current user location.

The 8motions server acts as a gateway server to the resources available on the map servers like Google Maps, Yahoo! Maps, Ask.com Maps or MSN Maps.

The user of the "GlobeTrotter" system does not have to update maps. As all map tiles will be downloaded over the Internet from the original servers, it is guaranteed that only up-to-date maps will be displayed.

The user can also use different additional functions for example move the map with a key, zoom in and out, switch to other map providers, remove or activate the scale bar and other features. Fig. 4 demonstrates the described functionality of the "GlobeTrotter" system.

The "GlobeTrotter" system receives the GPS position data over a logical serial port connection. A logical serial port is defined as a logical connection through which bytes are transferring serially and is defined within the underlying operating system. As the emulator can only receive data over COM port 1 or 2, the GPS connection for the execution of the "GlobeTrotter" application has to be established via COM2. Fig. 5 illustrates the establishment of the serial port connection to the GPS device.

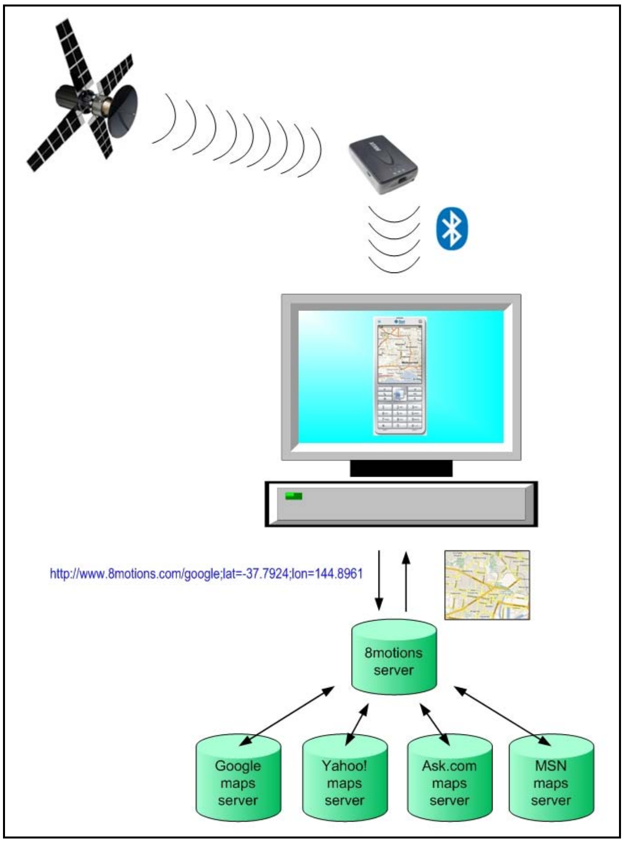

Fig. 4: Functionality of “GlobeTrotter“

```
// Serial connection to GPS device
private CommConnection connection;
// Serial GPS connection URL
private String url = "comm:COM2";
// Input reader to read data sent by GPS device
private InputStreamReader reader;
/* Establishes a serial connection (specified
* in url) and opens an input stream.
* @throws IOException
* if error occurs while establishing serial
* connection or opening input stream
*/
private synchronized void connect() throws
IOException
{
connection = (CommConnection)Connector.open(url);
connection.setBaudRate(19200);
reader = new InputStreamReader
(connection.openInputStream());
}
```

Fig. 5: Serial port connection establishment

The developed system reads in single NMEA 0183 sentences delivered by the GPS receiver and analyses them.

The latitude and longitude format received from the GPS receiver is in degree format of the World Geodetic System 1984 (WGS84). WGS84 is the geodetic reference system used by GPS and has been specified in 1984. Due to the fact, that all maps like Google Maps, Yahoo! Maps or Ask.com Maps use decimal longitudes and latitudes, the data conversion has to be done.

8. Implementation of the “GlobeTrotter” system

In the following, the implementation of the “GlobeTrotter“ system will be illustrated as an activity diagram. A complete flowchart with possible exceptions and use cases is presented.

Fig. 6 illustrates the approach of the developed system during localization of the current user position while Fig. 7 presents the proceedings during displaying map on the screen of the emulator.

Both activity diagrams summarize the functionality of the “GlobeTrotter“ system.

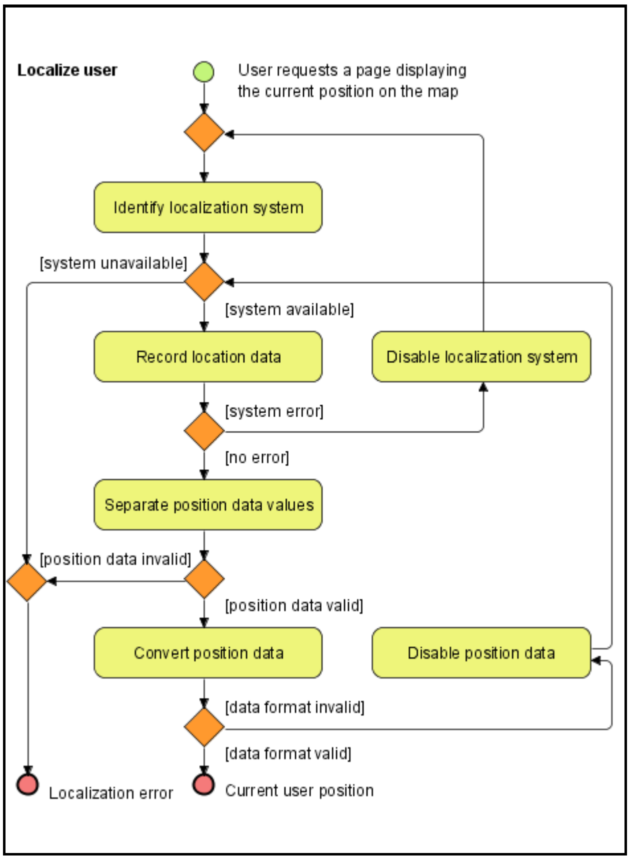

Fig. 6: “Localize user” activity diagram

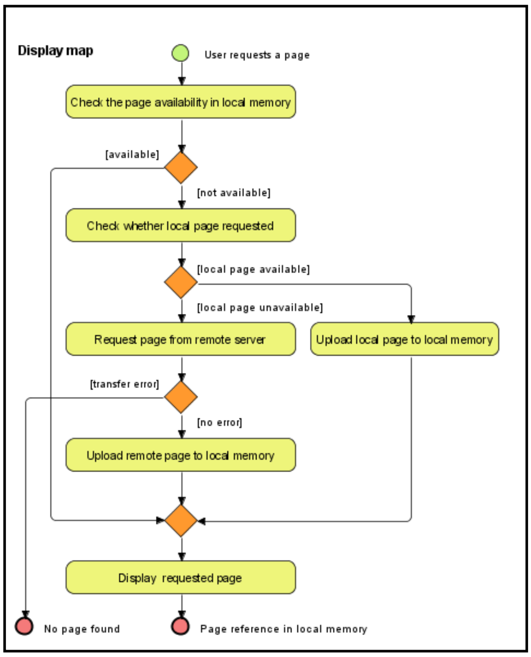

Fig. 7: "Display map" activity diagram

9. "GlobeTrotter" on a mobile device

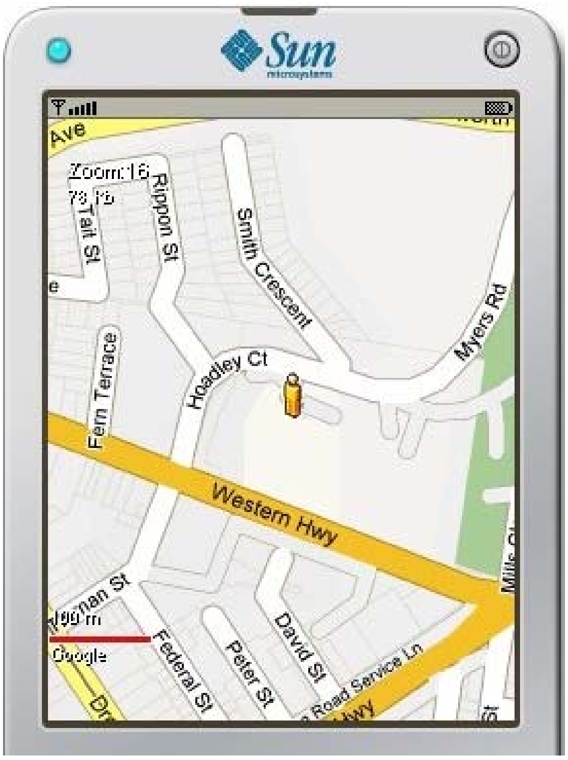

Fig. 8: Google Maps in zoom 16

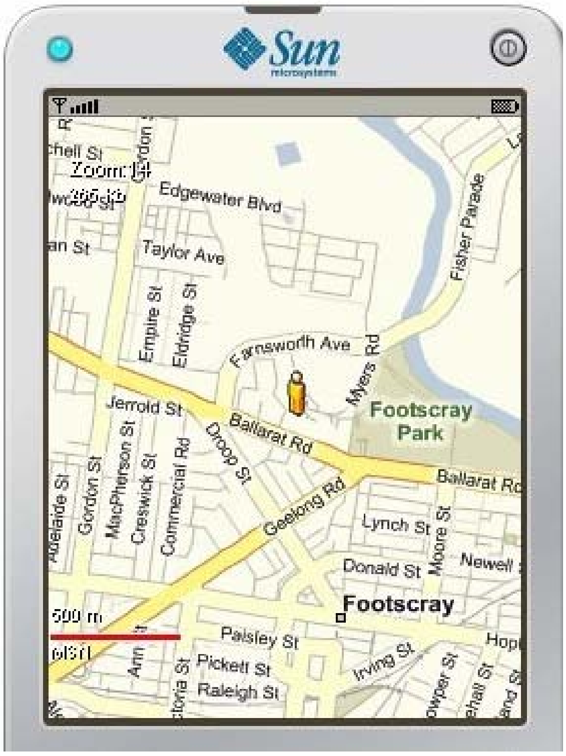

Fig. 9: MSN Maps in Zoom 14

Google Maps zooms range between 0 and 17 where 0 is the high-angle view and 17 is the most detailed view. Both Fig. 8 and Fig. 9 illustrate the user position on the Footscray Park campus of Victoria University in Melbourne (Australia) using Google Maps or MSN Maps in different zooms.

10. Conclusions and future work

"GlobeTrotter" based on the J2ME architecture and its components is successfully designed and developed. The entire system is fully tested and executed using J2ME Wireless Toolkit. With help of a GPS receiver, the user position can now be automatically determined and displayed by the "GlobeTrotter". All maps used in "GlobeTrotter" are downloaded in real-time from the original servers via 8motions.com server. The "GlobeTrotter" supports the transmission and the display of maps from internet mapping servers such as Google Maps, Yahoo! Maps, MSN Maps and Ask.com Maps.

The abstract design of the developed software is proved to be positive. The object-oriented approach used during the implementation guarantees an easy and quick extensibility of the functionality. In consequence of the modularity of the system architecture, created framework can be used for future applications. The separation of the content to be displayed on the screen and the technology it is based on allows the re-utilization of this system even in another context.

The achievement of the successful implementation of "GlobeTrotter" is enormous. It sets a good example of a way to integrate GPS with Internet mapping services as well as provides an innovative option to develop location-based systems in future mobile applications. It would be much appreciated to be able to use a system which can display road maps, weather maps, state maps, biking or hiking maps all in one.

References

[1] Isakow, A. and Shi, H. "Review of J2ME and J2ME-based Mobile Applications", International Journal of Communication and Network Security, Vol. 8 No. 2, pp. 189-198.

[2] Ortiz, 2004a, A Survey of J2ME Today, Sun Developer Network (SDN), viewed 13 August 2007, <http://developers.sun.com/mobility/ getstart/articles/survey/>.

[2] Ortiz, 2006, Summary of CLDC-Based Profiles, Sun Developer Network (SDN), viewed 23 August 2007, <http://developers.sun.com/ mobility/midp/ttips/cldc/>.

[4] The Boston Consulting Group, 2000, Mobile Commerce, viewed 25 October 2007, <http://www.bcg.com/publications/files/M-Commerce_Nov_2000_Summary.pdf>.

[5] El-Rabbany, 2006, Introduction to GPS: The Global Positioning System, Artech House, Electronic Resource, viewed 31 October 2007, <http://library.vu.edu.au>.

[6] Rosenberg, n.d., Global Positioning System: Eight Things You Need to Know About GPS, About.com, viewed 20 September 2007, <http://geography.about.com/od/geographictechnology /a/gps.htm>.

Ms. Anna Isakow is a final year Business Computer Science student at University of Applied Sciences Regensburg in Regensburg, Germany. She was supervised by Dr. Hao Shi as an international exchange student at Victoria University from August 2007 to November 2007 and successfully completed her minor thesis with topic "J2ME – specification, development and operation using the example of smart devices."

Dr. Hao Shi is an Associate Professor in School of Computer Science and Mathematics at Victoria University, Australia. She completed her PhD in the area of Computer Engineering at University of Wollongong and obtained her Bachelor of Engineering degree at Shanghai Jiao Tong University, China. She has been actively engaged in R&D and external consultancy activities. Her research interests include p2p Network, Location-Based Services, Web Services, Computer/Robotics Vision, Visual Communications, Internet and Multimedia Technologies.