IMOGA: An Architecture for Integrating Mobile Devices into Grid Applications

Ersan Öztürk

Istanbul Technical University
Institute of Science and Technology
Ayazaga Campus 34469 Istanbul Turkey
ozturker@itu.edu.tr

D. Turgay Altılar

Istanbul Technical University
Institute of Science and Technology
Ayazaga Campus 34469 Istanbul Turkey
altilar@itu.edu.tr

Abstract

Integrating mobile devices into Grid technologies and server applications can give ability to command power of supercomputers with a mobile device on one hand and can allow big applications to reach important data anywhere, anytime, on the other. IMOGA is planned to be an example to gather and share data that can be collected by ubiquitous mobile devices which can employ different kind of sensors such as Global Positioning System (GPS), temperature, health monitoring and pollution. In this project location and speed information that is produced by GPS enabled mobile devices such as mobile phones, is used. The developed client application running on mobile devices located in vehicles, such as the mobile phone of the driver, sends location and speed information to the server application in short time intervals via GPRS in the forms of Extended Mark-up Language (XML) like messages. The developed server application, which is preloaded with the highway coordinates via files in Geographic Data Files (GDF) format, locates the street that the vehicle is moving along and the received speed information is recorded along with a timestamp. A display application has also been implemented to calculate average of speeds that any vehicle may have at that very moment and post it on the Internet and WAP. If there is no actual data, i.e. there is no vehicle moving on a specific street, statistical data is utilised to produce such information.

1. Introduction

Grid Computing is an emerging concept which changes conventional way of thinking in computer technology. Currently researchers on Grid Computing are concentrated on pooling the resources of high capacity processing and storage units with strong visualization centers to solve hard scientific problems[1]. We believe that integrating mobile devices into Grid technologies can enable a simple mobile device to command even supercomputers, and further more applications running on supercomputers can acquire data instantly regardless of geographic location and time. The only requirement is a wireless communication support.

According to Metcalfe's law as the number of devices and users increases, grid-based resources become more valuable[2]. Participating mobile devices can increase the number of users to a number which can not be compared to today's population in the Grid so that Grid based resources become more valuable.

In this paper we suggest that a novel architecture called IMOGA (Integrating Mobile Devices into Grid Applications) to address the issues of integrating mobile devices into a Grid system and sharing and producing information in a cooperative manner. There are two different categories of wireless devices. Laptops are in one category; Personal Digital Assistants (PDA), mobile phones are on the other. In [3], the words *interlocutor* for the first category and *minions* for the second, have already been coined. The same terminology will be used throughout in this paper. One interlocutor and various numbers of minions form a cluster. The interlocutor acts as a proxy and represents minions to the Grid so that the system can be assembled by utilizing minions possessing even limited communication and computation power. The interlocutor has to be powered with a certain amount of computational, networking and storage resources given that these resources are not just the ones it has, but also the available resources in the minions connected to interlocutor. With these abilities interlocutor can run Grid middleware or have a mechanism to connect with a middleware (Figure 1).

Although PDAs and mobile phones come with different hardware and softwares, they will be considered in the same category (i.e. minions) since they share the same communication protocols. These devices will be connected to interlocutors in order to participate in Grid since they can not run a middleware on them. Note that the well known Grid portal called Globus requires substantial amount of memory and storage, which eliminates possibility of minions to run a Globus API[4].

Mobile phones are the basic minions. They are devel-

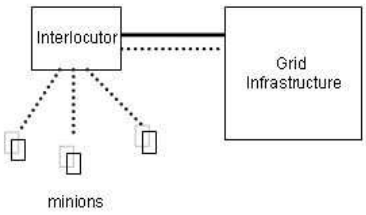

Figure 1: IMOGA architecture

oping very fast that special operating systems such as Symbian and Windows Mobile are being developed. Such operating systems can run on devices from different manufacturers. A common feature of these devices is the support for Java 2 Micro Edition (J2ME) which enables developers to have a common platform for their applications. Besides cameras and microphones mobile phones increasingly employ new devices such as GPS receivers and accelerometers. Moreover there is a possibility of attaching sensors to the mobile phones which can supply data on temperature, personal health, environmental pollution etc. These sensors can be easily connected via Inter Integrated Circuit (I^2C) protocol [5] or Bluetooth®. And about the most important limitation of mobile devices, battery consumption, developments in lithium polymer replacements for lithium ion show promise in this field [3]. Being able to collect data from anywhere, anytime, shall give us a big area to develop interesting and useful applications to be used in Grid applications. The IMOGA traffic application is based on mobile phones with GPS capability on board.

There are some applications developed using Wireless Grid even these applications are just far beyond the potential usage. Distributed Ad-Hoc Resource Coordination (DARC) lets devices with no prior knowledge of each other collectively record and mix an audio signal such as a concert, speech, lecture, or emergency event[6]. In the research carried out in Helsinki Institute of Physics called Gridblocks, healthcare application is developed that can detect the movement of lungs and heartbeats with 90% certainty within 50 cm distance of human body to produce valuable information of pulse and breathing rate[7]. The AKOGRIMO project is a European funded project aiming to architect and prototype Next Generation Grid based on Open Grid Services Architecture (OGSA) which exploits and closely co-operates with evolving Mobile Internet infrastructures based on IPv6[8]. The applications that are worked on or planned to be worked are e-Health, e-Learning and Crisis management. The GridLab project is one of the biggest European researches in the development of application tools and middleware for Grid environments. They are working on a workpackage to develop theoretical scenarios and their implementation for using mobile devices to enhance the working environment and efficiency of application users[9].

Figure 2: CIM-TR architecture

In this paper the IMOGA architecture is tested against an application that collect data ubiquitously and produces information for the very same data collection nodes. In this application (cooperative information management for vehicle traffic flow CIM-TR) minions transfer their location and speed information to the interlocutor and interlocutor processes the received data and provides traffic information to data providers as well as the public. CIM-TR architecture is given in Figure 2.

2. Methodology and Implementation

The implementation was composed of three main parts: client application that runs on minions, server application and display application runs on the interlocutors.

Given that three applications running over two platforms of which one is mobile communication between client and the server will be introduced first. The connection type is called three tier mobile web service[10]. The three tiers are client, web application containing servlet, and server. The client running on mobile device communicates with the middleware servlet using a proprietary communication protocol. The servlet and server communicate using Simple Object Access Protocol (SOAP) XML messages. The code necessary for a mobile client has a footprint that can be as small as 3 kilobytes. Therefore it can even be deployed on low end Java enabled mobile devices. In three tier communication system, the network bandwidth used in calls from the client to the server is known to be efficient [10], and the overhead produced for our application, i.e. CIM-TR, is negligible. The XML messages are sent to the server over the General Packet Radio Access (GPRS) protocol.

Client application is a Java MIDlet which runs on mobile devices with J2ME Mobile Information Device Profile (MIDP) support and GPS receiver hardware. Application acquires date, latitude, longitude and speed value from GPS

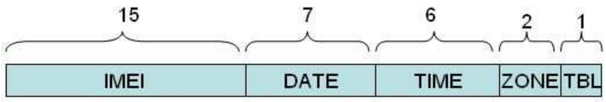

Figure 3: The fields in the header

receiver with JSR 179 Java location API. Together with these data, International Mobile Equipment Identity (IMEI) unique identification number is also passed to the server. The acquired data are sent to the server in SOAP XML messages. SOAP XML messaging protocol provides various data formats to send this data. Since minimization of data transfer cost is essential due to GPRS charging policy of GMS operators, we have done a number of experimental runs to choose the data format that will cost less. A string with fifty characters, one double, or four integers are sufficient to transmit a single location and speed data. The results of the test runs indicated that transferring data in the forms of strings lead to the minimum cost with maximum flexibility. Therefore it is decided to use the mechanism given below that packs data as two strings.

The message is composed of two parts: header and location_data. The header has a fixed length whereas location_data is a variable.

The header is composed of the following fields as shown in Figure 3.

IMEI: 15 characters. It is the identification number for all cellular phones which is unique for each device. It will be used as an identification for messages also in our communication.

DATE: 7 characters. Date is in DDMonYY format

TIME: 6 characters. Time is in HHMMSS format

ZONE: 2 characters. Timezone is important for international speed calculations. It is also required in countries using different timezones such as USA.

TBL(Time Between Locations): 1 character. Since there can be more than one location information in one message, it is required to show the time interval between these location readings. On server side it will be used to calculate the exact time for each location reading. It shows time interval in seconds.

Location_data is composed of the following fields(Figure 4).

NOL (Number Of Locations): 1 character. It shows the number of location readings that will be sent in a single location_data. It gives us the flexibility on number of readings to be sent in a single message.

LATITUDE: 9 characters. It is the latitude of the mobile device. By using the standard precision representation 9 characters is shown to be sufficient to have a precise location information in the form of DD°MM'SS.SS".

LONGITUDE: 9 characters. It is the longitude value of the mobile device. Similar to the latitude parameter 9 characters would be sufficient for representation.

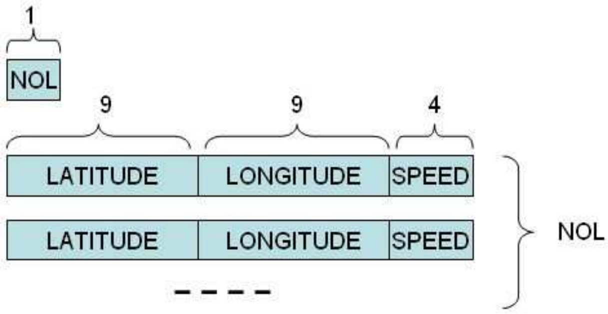

Figure 4: The fields in the Location_data

SPEED: 4 characters.Most of the GPS receivers in the mobile devices automatically produces a speed value. For those of which does not produce will require a simple mathematical calculation to produce this value.

TBL and NOL parameters provides us flexibility over cost and efficiency. Values of these parameters are fixed in our experiments but they can be programmed to be adaptive according to the speed of the mobile device.

The client application has also a basic interface. This interface shows longitude and latitude in degree, minute, second format(Figure 5). As the GPS receiver in mobile device returns latitude and longitude values in double format, the application converts these values to degree, minute and second format. The client application also indicate speed and the state of GPS receiver that can be either available, out of service and temporarily unavailable. Since update interval of locations and the number of location information in one message are also variables for cost estimation they are displayed in via the interface. Furthermore the user may adjust these values for cost reduction. Every time the application sends a message, "MESSAGE SENT" indication is displayed on the client interface.

Server application runs on a Sun Java System Application Server 9. which is open to the internet. The server application has to be connected to the internet since the client application messages are delivered through internet via GSM operator. This application works on interlocutor and is designed to communicate with a Grid middleware. Database operations are implemented locally through MySQL. The main purpose of the server application is to locate the street that the vehicle is moving along and record the received speed information along with a timestamp for that street.

After researches on Following a survey digital map formats and standards such as GDF, Federal Geographic Data Committee (FGDC)[11], ISO 19110, ISO 19115 [12]; it was decided to implement a mapping system based on GDF.

Figure 5: Client Interface

GDF is a European standard that is used to describe and transfer road networks and road related data. It was drawn up by Comite Europen de Normalisation (CEN) in co-operation with digital map providers, automotive and electronic equipment manufacturers.

GDF has a three-level structure: level 0 is a common Geographical Information Systems (GIS) topology description that everything has been described by nodes, edges and faces. Level 1 contains simple features such as road elements, rivers, boundaries, and signposts. At level 2 the simple features are aggregated to a higher-level feature.

The road database tables in CIM-TR are created according to GDF specifications [13] with some variations. The main tables and their fields that are created for different levels are given in Table 1,2,3.

Table 1: Main tables for level 2

Location
Location_ID
IMEI
location_date
location_time
location_timezone
latitude
longitude
speed

There is a detailed procedure to decide if a location is on a road preloaded to the database. Two new tables are used for these procedures: User_last_location and Road_speed (Table 4). User last location keeps the last location of the mobile if at the end of the procedures mobile seems to be on one road or on two roads. The latter case is possible on two sided roads when the user drives close to opposite side. If mobile seems to be on one road the road is recorded as the *Certain Road*, whereas if it seems to be on two roads, the two roads are recorded as the two *Candidate Roads*.

When application receives a new location message, after decomposing the messages it checks if the same device sent data in X seconds where X is a variable with default value of 20. If yes, it checks if the mobile was on *Certain Road* or two *Candidate Roads*. If it is on *Certain Road*, then it is checked if it is on the same road. This saves time. Otherwise the location is treated as first data from user. If it is on two *Candidate Roads*, old location data is taken into calculations along with new location to decide on the direction of the mobile and locate the mobile on one *Certain Road*. IfOnLineWithDirection procedure is given below.

Table 2: Main tables for level 1

Lines	Points
Line_ID	Point_ID
Edge_ID	Node_ID
From_point_ID	Point_name
To_point_ID	
Line_name	
Line_width	

Table 3: Main tables for level 0

Edges	Nodes	XY_coordinates
Edge_ID	Node_ID	XY_ID
XY_ID	XY_ID	latitude
From_node_ID	Face_ID	longitude
To_node_ID		
Left_face_ID		
Right_face_ID		

Table 4: User_last_location and Road_speed tables

User_last_location	Road_speed
IMEI	Road_speed_ID
Last_date	Date
Last_time_ID	Time
Last_latitude_ID	Road_ID
Last_longitude	
Cand_roadEle_ID1	
Cand_roadEle_ID2	
Cert_road_ID	

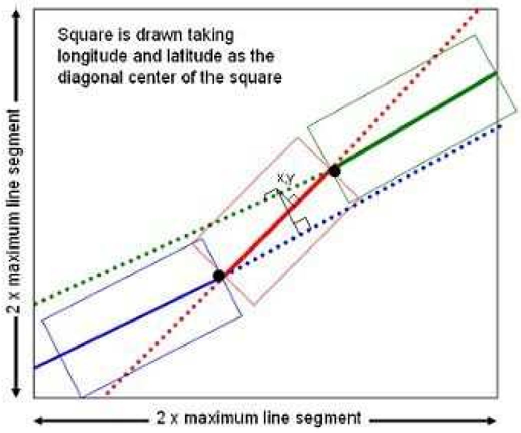

Figure 6: IfOnLine procedure illustration

```
IfOnLineWithDirection
find the equation of the line passing through two points
find the distance from location to the line
find the closest point of the line to the location
if the closest point is on the line segment
    dis_oldtostart = distance from old
    location to the starting point
    dis_newtostart = distance from new
    location to the starting point
    if the distance is smaller than the line width
        if dis_newtostart is greater than dis_oldtostart
            the location is on the line and
            it has moved in road direction
        else
            the location is on the line and
            it has moved opposite to road direction
    else point is not on the line
else point is not on the line
```

If the client application has not sent any message within a preset period of time recently or the device is not on the *Certain Road* or the two *Candidate Roads*, the received location information is treated as the first data coming from that particular mobile device. A virtual square is drawn with an edge twice as long as the maximum line segment where the received location coordinate stays at the center of the square. It is checked whether the location is on the lines which have any part inside this square(Figure 6). Depending on the result it can be declared as moving along on the *Certain Road* or the two *Candidate Roads*.

In these procedures if the location comes out to be on a *Certain Road*, a new speed information is recorded with a timestamp in the Road_speed table.

Display application is a Java applet that connects to the database to retrieve user locations and speed information and display it to the public.

A decision has to be made when there is no current speed data for a particular road. One of the important part of the CIM-TR project was to decide how to calculate the traffic on a road especially if there is no speed data at current date and time. A decision has to be made when there is no current speed data for a particular road. If there is a feed in about a particular road within less that half an hour it may be accepted and published as the current data. However the use of statistical data becomes essential if there is no such data available.

It was decided to categorize statistical data and use them with different coefficients when calculating the traffic information. Five different set of speed information are returned from database starting from the least valuable to most valuable one. These sets are listed below.

1. Average of speeds everyday in that hour(i.e.16:49-17:49)

2. Average of speeds in that weekday in that hour(i.e. Thursdays 16:49-17:49)

3. Average of speeds in one hour(i.e.Today 16:49-17:49)

4. Average of speeds in half an hour(i.e.Today 17:19-17:49)

5. Average of speeds in 10 minutes(i.e.Today 17:39-17:49)

The highest number in the order indicates the dominance of the figure that we have to consider. Moreover the data in the lower numbered average includes the average of the higher ones. In another point of view if there is no available data for a specific level, there is no data for higher levels. Since data in set 5 is considered as instant if there is data available in set 5 then it contributes 80% to the result and the rest 20%. Otherwise higher level sets contribute 60% whereas lowers contribute 40% totally. The contribution of sets according to the availability of data in them given in Table 5.

Table 5: Percentage contribution of sets to the result

Data in set(s)	1	1,2	1,2,3	1,2,3,4	1,2,3,4,5
1	100	40	16	6.4	1.28
2		60	24	9.6	1.92
3			60	24	4.8
4				60	12
5					80

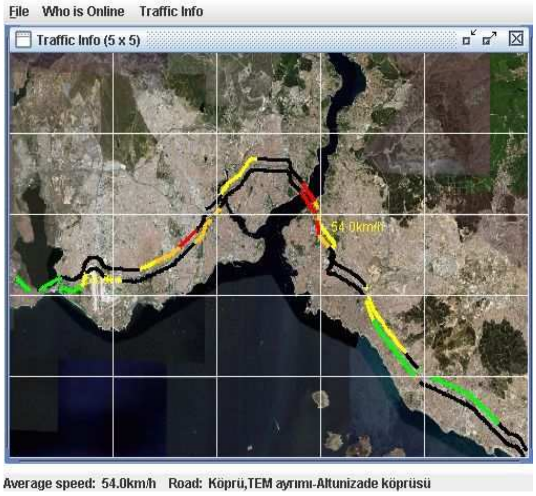

Figure 7: Display application interface

The roads are displayed in colors that are set according to the speed information returned from the database(Table 6).

Table 6: Display colors for road traffic

Traffic condition	Speed(km/h)	Color
No information		Black
Very heavy traffic	0-30	Red
Heavy traffic	30-50	Orange
Running Traffic	50-80	Yellow
Open traffic	80-	Green

Display application presents the speed information on a Istanbul map which is composed of 5 x 5 = 25 parts. Users can also see these parts in details. Mouse cursor can be moved on the road to see the name of the road and exact average speed on that road(Figure 7).

3. Results and Discussion

The IMOGA is developed to collect data from ubiquitous mobile devices and use in Grid applications. An application namely CIM-TR has been developed based on the IMOGA architecture. CIM-TR collects location and speed data from mobile devices integrated with GPS receiver. Initial results indicate that the IMOGA is a suitable architecture for ubiquitous grid applications. Mobile phones have been used as primary mobile devices in CIM-TR application since they are in use extensively and there is a significant trend to integrate GPS receiver in these devices. Besides their week computational power, GPS data produced by the mobile phones is the most valuable data that can be shared for the time being. Having the CIM-TR client installed on the mobile phone, location and speed data can be sent in SOAP based XML messages to the web service offered by the server application running on interlocutors. The developed communication mechanism enables to transfer data with minimum cost.

The geographic coordinates of the main highways in Istanbul are made available in the database. Traffic information on these highways are presented to the users on a map in a public web page. Visitors of the page can see the average traffic speed on the roads separated by junctions in every 3 - 4 km, on a map of different scales. The same information can be acquired by the mobile phones as well. However the graphical interface is not available yet.

The server application procedure that was used to find the road that the location information has been sent from has a success rate of 91%. This was calculated in a test by the rate of records that was entered to the road speed table to the total number of messages sent by test mobile in certain part of highway in certain amount of time. The analysis of the test results shows that there are three essential reasons having a loss rate of 9%. As it can be seen in Figure 6, roads are treated as aligned rectangles and the connection points of these rectangles yields small uncovered areas. Data coming from uncovered areas is the first reason to e mentioned. Second reason is that road widths are kept as lane numbers and Istanbul roads does not have the standard lane widths. The last but not the least reason is the sensitivity of the GPS receivers in location based services. But the GPS sensitivity of the chipsets that are used in the mobile devices are developing and highly accurate today which is increased to 2.5 meters of horizontal position accuracy[14].

Increasing the number of mobile devices that runs client application is the most important fact for the CIM-TR to perform better and produce more reliable results. Convincing users to participate in a public project for communal benefits is an essential problem. However the IMOGA may offer other services than traffic information that would help to convince them. In order to increase the use of CIM-TR, a side product has already been produced: vehicle tracking system. Users can register to the system with their IMEI numbers and track their vehicles live on map. Since users have to pay for their GPRS cost to the GSM operators, it is possible to make an agreement with operators to make discount for the participants, in return presenting the CIM-TR traffic information in their WAP or/and WEB sites. Triangular economic models between users, grid infrastructures and Internet service Providers (ISP) are hot research topics [3].

When there are several users on roads, the CIM-TR will supply sufficiently correct traffic data to the user. When there is no online users on a road, information produced with statistical data has to be displayed. The computation over the statistical data is implemented in such a way that recently acquired data dominates the produced result Also note that statistical information based on higher valued sets is more reliable than lower valued ones as shown in Table 7. A difference between actual and calculated value is inevitable when calculation is done excluding most valuable set. However this difference is not significant since the coefficients for different sets are chosen according to observations in Istanbul traffic and database. Table 7 gives an example from database records for a road and the results that would be displayed according to the sets taken into calculations. The instant speed value was 21.0 in this test and as it can be seen from table the results displayed gets closer to the instant value as there is data in more valuable sets.

4. Conclusion and Further Research

The IMOGA has successfully established the mechanism to collect data from mobile devices and process them. The CIM-TR application was important to show how data, that is collected from these ubiquitous objects, can be processed and made open to the participants and public. It can be a baseline for sharing resources between huge numbers of mobile devices and high capacity Grid infrastructures which can produce applications on e-health, e-crisis management and e-government.

There are several areas that the IMOGA can expand. The most important implementation to be added to the IMOGA is to develop the mechanism that would connect server in interlocutor with Grid middlewares and share resources from ubiquitous mobile devices with Grid infrastructures. The communication between minions and interlocutor can go further that minions announce available services and data they can produce and interlocutor can keep an Universal Description, Discovery, and Integration (UDDI) directory for these services. It can check which minions are available in time intervals. Temperature, health monitoring and pollution sensors can be connected to the mobile phones as accessories via I^{2}C wired and Bluetooth®wireless technologies. I^{2}C communication mechanism with mobile phones is worked on for the IMOGA currently. Implementing health monitoring sensors and using in Grid applications seems to be a very promising research area.

Table 7: Displayed results according to data sets

Results	Set 1	Set 2	Set 3	Set 4	Set 5
In set value	15.7	17.2	20.2	22.3	21.1
Calculated value	15.7	16.6	18.8	20.9	21.1

References

[1] F.Berman,G. Fox, T. Hey, 2003. "The Evolution of the Grid" in Grid Computing - Making the Global Infrastructure a Reality, pp. 65-100. D.D. Roure,M.A. Baker, N.R. Jennings, N.R. Shadbolt, John Wiley and Sons Ltd,2003.

[2] L.McKnight, J.Howison, S.Bradner, "Wireless Grids Distributed Resource Sharing by Mobile, Nomadic, and Fixed Devices", *IEEE Internet Computing*, 2004.

[3] T.Phan, L.Huang, C.Dulan, "Challenge: Integrating Mobile Wireless Devices Into the Computational Grid", *MOBICOM'02*, Atlanta Georgia, USA,2002.

[4] J.Hwang,P.Aravamudham, "Proxy-based Middleware Services for Peer-to-Peer Computing in-Virtually Clustered Wireless Grid Networks", http://wirelessgrids.net/docs/CCCT03_Hwang_T983ST.pdf (as of 04/05/2007)

[5] I^{2}C specification http://www.standardics.nxp.com/literature/-books/i2c/pdf/i2c.bus.specification.pdf (as of 04/05/2007)

[6] M.Tuisku, "Wireless Java-enabled MIDP devices as peers in Grid infrastructure" , First European Across Grids Conference, Santiago de Compostela, Spain, 2004.

[7] Gridblocks project page http://gridblocks.sourceforge.net (as of 04/05/2007)

[8] Akogrimo project page http://www.mobilegrids.org (as of 04/05/2007)

[9] Gridlab project page http://www.gridlab.org (as of 04/05/2007)

[10] Java web page http://www.java.sun.com

[11] Standard for Digital Geospatial Metadata 2.0 http://www.fgdc.gov/standards/projects/FGDC-standards-projects/metadata/base-metadata/v2_0698.pdf (as of 04/05/2007)

[12] ISO standarts page http://www.iso.org

[13] Geographic Data Fies Specification 3.0 http://www.ertico.com/en/links/links/gdf_-_geographic_data_files.htm (as of 04/05/2007)

[14] GPS chipset producer Sirf Technology web page http://www.sirf.com (as of 04/05/2007)

[15] M. Kahveci, F. Yldz, *GPS Teori Uygulama (GPS Theory and Applications)* Nobel Publishing, 2005.

[16] M. J. Yuan *Enterprise J2ME: Developing Mobile Java Applications* Prentice Hall, 2005.

[17] K. Topley, *J2ME in a Nutshell* O'Reilly and Associates Inc., 2002.

[18] L.C. Larijani, *GPS for everyone* American Interface Corp. New York, 1998.

[19] I. Foster, C. Kesselman, S. Tuecke, ”The Anatomy of the Grid”,*Intl J. Supercomputer Applications*, 2001.

[20] I. Foster C. Kesselman *Grid 2*,Elsevier Inc., 2004.

[21] L.McKnight, J.Howison, ”Towards a Sharing Protocol for Wireless Grids”, *CCCT '03*, Orlando, Florida, USA, 2003.

[22] M.Tuisku, J.Karppinen, T.Niemi ”Java Enterprise Grids using JBoss middleware”, Helsinki Institute of Physics, CERN.

[23] P. Grabowski, B. Lewandowski , ”Mobile-enabled grid middleware and/or grid gateways”, Grid Lab Project, 2001. (as of 04/05/2007)

[24] J. Hwang, P. Aravamudham, ”Middleware Services for P2P Computing in Wireless Grid Networks” *IEEE Internet Computing*, 2004.

[25] N. Apte, K. Deutsch, R. Jain, ”Wireless SOAP: Optimizations for Mobile Wireless Web Services” *14th World Wide Web conference*, 2005.

[26] A. Hampshire, ”Extending the Open Grid Services Architecture to Intermittently Available Wireless Networks” www.allhands.org.uk/2004/proceedings/papers/232.pdf (as of 04/05/2007)

[27] B. Hull,V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, S. Madden ”CarTel: A Distributed Mobile Sensor Computing System” *4th International Conference on Embedded Networked Sensor Systems*, 2006