# ENHANCING MOBILE PEER-TO-PEER ENVIRONMENT WITH NEIGHBORHOOD INFORMATION

Arto Hämäläinen and Jari Porras
*Lappeenranta University of Technology*
*Laboratory of Communications Engineering*
*P.O. Box 20*
*53851 Lappeenranta*
*Finland*
*{Arto.Hamalainen, Jari.Porras}@lut.fi*

**ABSTRACT**

In this paper some enhancements to the performance of mobile peer-to-peer environment are presented. The work is based on our implementation of mobile peer-to-peer communication environment, PeerHood. PeerHood offers direct communication between mobile devices by using Bluetooth, WLAN or GPRS communication technologies. PeerHood is enhanced by allowing devices to exchange their neighborhood information thus extracting better view of the environment. This information can both bypass problems caused by incomplete device discovery and provide information about devices outside the immediate vicinity.

**KEYWORDS**

Personal trusted device, peer-to-peer, neighborhood, ad hoc networking, device discovery, service discovery

## 1. INTRODUCTION

Current trends in networking consist of wireless environments and mobile devices. These devices are used both in infrastructure-based and ad hoc networks. Another widely discussed topic is the ubiquitous computing. Until recently, all these concepts have been just ideas because non-existent devices and network technologies. However, recently the utilization of wireless environments like 802.11 wireless LANs and Bluetooth wireless technology has rapidly increased.

Device and especially service discovery is fundamental part for mobile ad hoc environment. Usually presumed roaming and mobility prevent the use of static directory-based service and device discovery and suggest that dynamic service discovery technology could be more suitable. Generally these dynamic service discovery protocols are used to advertise services on own device and search services on remote devices. Each device should have a client part which searches for services on other devices and a server part, which responds service discovery requests originating from other devices [1].

Several service discovery protocols have been developed, but mainly for infrastructure-based networks. These protocols have originated from and developed by both standardization organizations and industrial consortiums. Internet engineering task force has developed Service Location Protocol (SLP), which provides a scalable framework for the service discovery [2]. SLP is intended for networks which have cooperative administrative control. Therefore its usability for ad hoc networking must be inspected before adoption. Sun's Jini, Microsoft's Universal Plug-and-Play and Bluetooth Service Discovery Protocol are examples of industry-originated service discovery protocols. Bluetooth SDP is the only one of these which is targeted directly to mobile wireless network. [1]

Communications in ad hoc wireless networks are often established in a peer-to-peer manner. Peer-to-peer environment implemented in mobile devices assists importing applications to mobile ad hoc networking. In this paper some enhancements to the basic operation of our mobile peer-to-peer environment, PeerHood, are presented. First the PeerHood environment and its structure as well as operation is explained. Then the neighborhood information exchange and ways to utilize it are examined. Finally some scenarios and results for the evaluation of the enhanced PeerHood are given.

## 2. MOBILE PEER-TO-PEER ENVIRONMENT

The development in mobile devices, communication technologies and networking paradigms during the last decade has been rapid. Computational power of the mobile devices has increased, new short range networking technologies has emerged and personal networking paradigm has evolved. Although each of the developments alone is significant the combined effect is even more remarkable. Personal information management as well as intelligent and possibly seamless connectivity is possible if considering a powerful mobile device capable of short range communications. By allowing peer-to-peer connections between devices (without network infrastructure) new possibilities will emerge. Currently peer-to-peer approach is mainly used in fixed networks but some approaches to apply it to mobile networks have been proposed. Peer-to-peer communication can be seen as one of the most promising communication paradigms of the future as it allows resource sharing in a flexible manner. It allows truly dynamic networking and is suitable for mobile networks as well. There exist several approaches for the personal networking ranging from proximity based approaches, e.g. Digital Aura [3] and Virtual device [4], to approaches based on personal interests, e.g. Personal Networking [5], I-centric communications [6] and Personal Distributed Environment [7]. We propose another approach, namely peer-to-peer neighborhood, PeerHood. PeerHood is an implementation of a personal area network (PAN) based on peer-to-peer paradigm in mobile environment [8]. PeerHood is built inside a mobile device, i.e. Personal Trusted Device (PTD). Our approach is based on the need of local services (proximity) and the use of them through different networking technologies (Bluetooth, WLAN and GPRS). **Figure 1** presents the idea of PeerHood. Our PTD, i.e. mobile phone in this case, detects other devices in its vicinity. The size of the vicinity, i.e. neighborhood, depends on the communication technology used for observing. **Figure 2** presents another case where the neighborhoods of devices are not of the same size. It should be noticed that some of the devices might not be visible for all the other devices.

Figure 1. PTD and its neighborhood.

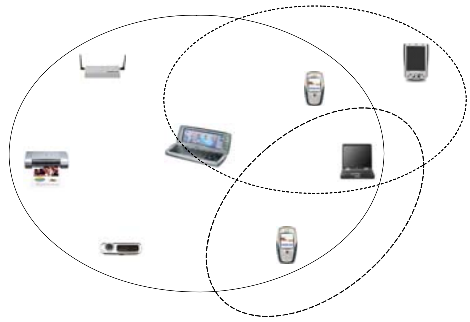

Figure 2. Neighborhoods of different devices.

The main goal of our PeerHood system is to provide a communication environment where devices act and communicate in a peer-to-peer manner. This means that devices communicate directly with each other without any centralized servers, i.e. PTD might communicate with all the devices in its neighborhood. In order to enable fast creation of the required ad-hoc type networks the immediate neighbors of a device are monitored and the gathered information is stored for possible future usage. This is presented in **Figure 3**. Information concerning devices or services within the neighborhood is stored into appropriate place in the PeerHood system. The second goal is to create a library that enables the usage of any supported networking technology via a unified interface so that the underlying networking structure is hidden from the applications point of view. As a direct consequence the application development time should be reduced because complex tasks like device discovery, connection establishment and error checking are handled by the PeerHood system. This library is shown as an interface to the environment managed by the PeerHood layer.

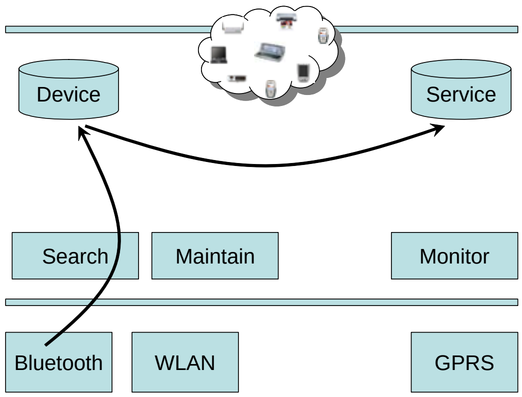


Figure 3. PeerHood layer in PTD.

In our approach the PTD is continuously sensing its neighborhood through different network technologies and it maintains the gathered information for the further usage. As other devices are observed they are added into the neighborhood of the PTD. The seamless connectivity is provided by the PeerHood layer through some basic operations i.e. search, monitor and maintain operations of the changing ubiquitous environment. As new devices and services are observed they are stored into neighborhood information tables. Thus the PTD always has up-to-date information concerning its environment. This information is provided for the application through the PeerHood interface. With PeerHood interface applications can list devices and services in their wireless neighborhood and connect to them when necessary. **Figure 4** presents an example of a file sharing application on PeerHood. This application shows the neighborhood (devices in upper box with their interfaces) and files within the selected device.

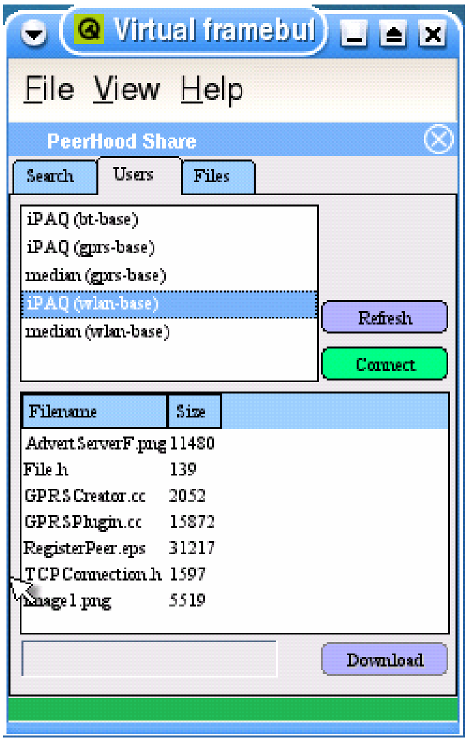


Figure 4. Mobile file sharing over PeerHood.

## 3. PEERHOOD IMPLEMENTATION

PeerHood operations have been implemented both in Linux based PDA and Symbian based mobile phone. In this paper we concentrate on the Linux implementation and its enhancements.

Linux implementation of the PeerHood environment is based on four elements, *PeerHood Daemon*, *PeerHood Library*, *Additional Components* and *PeerHood Applications*. Of these components the daemon and the library are mandatory for any application to work and thus will be presented in this paper. Additional components are optional middleware libraries that further extend the PeerHood's functionality and are presented in other papers [9]. **Figure 5** presents the components of the Linux based implementation of the PeerHood. The daemon is one independent process which takes care of locating other devices. The daemon implements device discovery using network specific *plugins* through the plugin-interface. Bluetooth, WLAN and GPRS plugins are available for the Linux PeerHood. The library interface provides all the PeerHood functionality to the application.

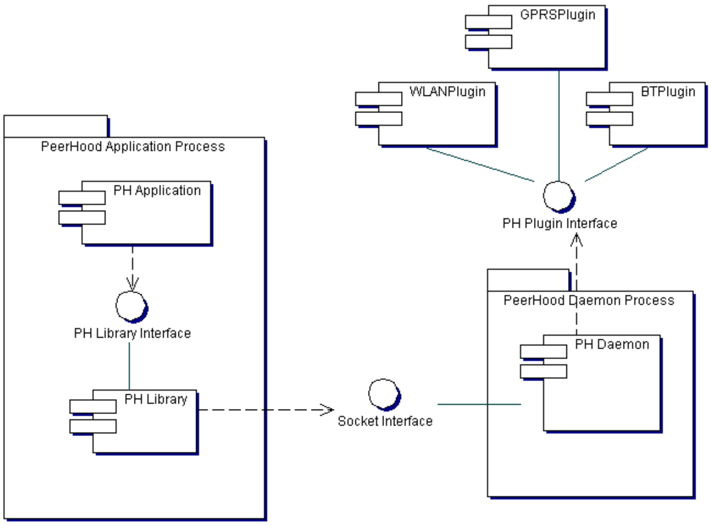


Figure 5. Components of the Linux PeerHood

Because the daemon and the library are independent components, PeerHood class structure is also divided into two parts. Important part of whole PeerHood structure is the abstraction between networking specific classes and interfaces. PeerHood allows developers to add new plugins which can be used by the daemon. To be able to provide this architecture, the whole technology-specific functionality must be contained in one class which implements the specific interface. This allows the core components to remain the same while adding new functionality using plugins. Daemon class diagram is illustrated in Figure 6.

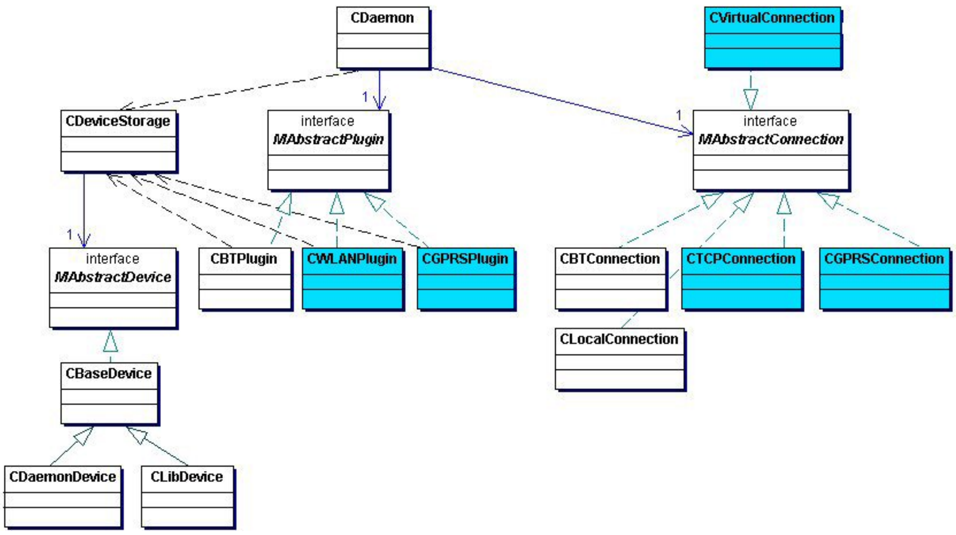


Figure 6 - PeerHood daemon class diagram

The main issues towards the goals of this paper are in the management of the neighborhood. This means that the functions taking care of finding and monitoring devices in the environment need to be studied. To find and keep track of devices in the neighborhood, PeerHood uses different device discovery functions of different network technologies. While using Bluetooth connections PeerHood can utilize its service discovery protocol, for WLAN or GPRS some specific methods has been implemented for discovering devices. These device discovery functions are not always the most efficient methods, i.e. the frequency of inquiries might degrade the performance of the whole system. In this paper we have focused to handle this problem with the neighborhood information exchange approach. By adding some neighborhood information into the inquiries the performance of neighborhood information management is improved.

## 4. NEIGHBORHOOD INFORMATION EXCHANGE

Mobile ad hoc and infrastructure-based networks have different requirements and facilities for service discovery. Peer-to-peer mobile networks are generally formed in an ad hoc manner. However, some of the devices of an ad hoc network can also be connected to an infrastructure-based network or their environment in an ad hoc network is changing slowly. Therefore they likely have more comprehensive information about the devices around than passing mobile devices. For this kind of situations where mobile ad hoc networks meet networks based on fixed infrastructure, exchange of neighborhood information may turn out to be valuable element of service discovery.

Neighborhood information exchange has been integrated to PeerHood operation. Neighborhood information is gathered from other PeerHood devices and it's used together with regular device discovery to provide constant view of devices nearby and further away. A message sequence chart for neighborhood information exchange can be seen in **Figure 7**.

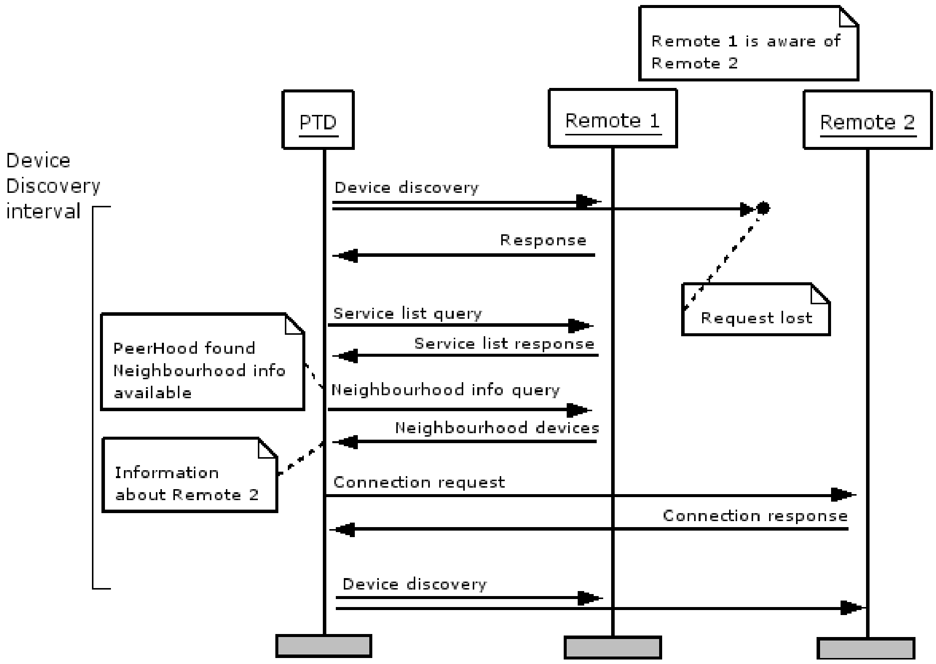


Figure 7 - Message sequence chart for neighborhood information exchange

In this example, *Remote 1* provides neighborhood information exchange service to other devices. Mobile Personal Trusted Device *PTD* is sensing its neighborhood with regular *Device discoveries*, which are performed at certain interval. Without neighborhood information exchange, failure to discover a device offering a desired service, in this case *Remote 2*, could postpone connection to service until the next device discovery. By using neighborhood information exchange, however, *Remote 1* provides the requesting *PTD* all the needed information about the *Remote 2.*

Neighborhood discovery is carried out by the PeerHood daemon class, whose structure is presented in Figure 5. Daemon is the component in the PeerHood, which takes care of locating other devices and provides applications with an interface to connections. Daemon also acts as a service discovery server, advertising services belonging to PeerHood applications running on a device. It uses different plug-ins (Bluetooth plug-in, WLAN plug-in and GPRS plug-in) through the PeerHood plug-in interface. The clients can connect to a daemon and send requests to it through a socket interface provided by the PeerHood library. Commands include listing of found devices and services, registering a new service and monitoring a connection. Although neighborhood information inquiry in general is a common operation for all network technologies, it is implemented within plug-ins alongside co-existing device discovery functions. Discovering available PeerHood services or neighboring PeerHood devices from another PeerHood device can be done consecutively or separately.

If a device gathers information about neighboring devices and wants to provide it to other devices also, the flag for neighborhood information exchange service is registered to PeerHood. When a device finds the neighborhood information flag on a remote device using service discovery, it requests the neighborhood information from it. The remote device then compiles the list of neighboring devices and services and sends the list to the requesting device.

Device information received by neighborhood information inquiry is stored to the device storage (CDeviceStorage) just like information received by regular device discovery methods. Along with regular information, address of device, where the neighborhood information was received, is stored. This way the connection to a service can be established through the intermediate device, if possible.

## 5. USAGE SCENARIOS

A few usage scenarios are designed for the neighborhood information. In a general level in these scenarios neighborhood information is used both to provide more comprehensive knowledge of the neighborhood and to gather information about devices outside the immediate neighborhood.

## 5.1. Personal networking outside the immediate vicinity

Personal networks are formed around a person and they consist of devices which are most likely to be used by him in his everyday routines [5]. Personal network is not limited to nearby devices within the reach of wireless short-range technologies, but may be extended via Internet or multi-hop wireless network to another set of devices. The use of neighborhood information provides information about distant devices. If an interesting device or service is found using neighborhood information exchange, then a connection could be established either directly using long-range network technology or via one or several intermediate devices, which may be the same devices that forwarded the device information in the first place.

## 5.2. Exchanging detailed device information

Advanced mobile devices could be used in several purposes in addition to regular personal communications and information management. The processing power and storage capacity is increasing rapidly and their support for different communication is becoming more and more comprehensive. These factors combined with increasing popularity of these advanced mobile devices mean, that there's quite a lot processing power available around us in everyday wireless neighborhood. Surrounding devices could be used in parallel manner to solve large computational problems.

In addition to authorization and compatibility issues, the ability to survey available resources is important element in this kind of operation. Resources can be scanned by exchanging detailed device information between mobile devices. This information includes information about the processing power, storage capacity and available services available at each mobile device.

## 5.3. Dedicated service discovery

Another usage scenario is based on presence of certain devices in certain locations. For example at home or at work there’s a personal computer available practically always. Instead of running device discovery constantly itself, a mobile device may use a dedicated computer to do the task. When information about other devices is needed, the mobile device can connect to the PC and request the information. An advantage is gained by this method, if the PC has other network connectivity not found in the mobile device, is that the mobile device gets information about devices which are not reachable by its own discoveries. If an important service is found on a device for instance in another floor or department nearby, the mobile device user may connect to the service using the PC or walk closer to the service and use it with his mobile device. If the mobile device has the needed connectivity available but disabled due to power consumption or any other reason, it may be enabled and the found service connected.

Another advantage of this method is the avoidance of battery consuming inquiry process in the mobile device. Determining the power consumption of Bluetooth or other wireless devices is difficult if the exact mode of operation is not known. While constant transmitting and receiving could lead to an early drain of batteries, idle or sleep modes will make it last a lot longer [10]. Although an idle Bluetooth chip requires in active mode almost half of the current compared to the send or receive modes operation (50 mA), the use of low-power modes reduces the power consumption drastically (down to 60µA) [11]. While device discovery modes differ from technology to technology, their operating scheme is similar. A device transmits a request to the network and begins listening for responses. When this is done constantly, the utilization of low-power modes is not possible.

## 6. EVALUATION

Evaluation was done in a Bluetooth environment similar to one that is pictured in Figure 7. The objective of the tests is to indicate enhancement in PeerHood operation by introducing neighborhood information exchange. The measurements are shown in the Figure 8. Ten tests were carried out and measured for both with and without neighborhood information exchange. Time measured was the time between the launch the PeerHood daemon in PTD and discovery of two other PeerHood devices located in a same room. Other Bluetooth devices were also detected, but the detection of these two PeerHood devices was considered the criteria for a completed test.

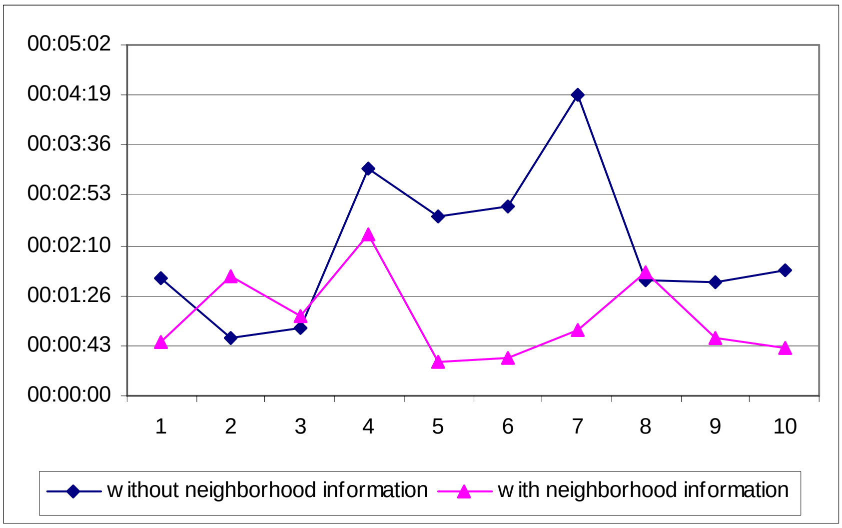


Figure 8 - Time taken to find two PeerHood devices with and without neighborhood information exchange

One of the other devices was a stationary device which was offering neighborhood information and had a 30 second interval between its own device discoveries. The other was acting as a mobile device with an interval of 10 seconds between device discoveries. This reflects the heterogeneity of devices in a real environment. The device and service discovery on the mobile device was often interfering our device discovery and the device remained undiscovered. However, the stationary device with a longer device discovery interval was found more certainly. As soon as it was found, the neighborhood information could be inquired and exchanged, and the mobile device was also discovered. The average time taken by the discovery was 77 seconds with and 129 seconds without neighborhood information, so these results show that neighborhood information enhance the operation in this kind of environment. Furthermore, implementing advanced fixed and mobile device roles could improve the operation even more.

## 7. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a mobile peer-to-peer based environment, PeerHood, and proposed some enhancements to its operation. These enhancements allow us a faster gathering of the neighborhood information and thus improve the whole environment.

Exchanging information like device addresses, names and service descriptions causes also worries about the security of the system. A suitable solution could be implementation of a generalized authentication module, which is used by other services and applications. This way, authentication policies for each service could be managed in a one module.

## REFERENCES


[1] Golden G. Richard III: Service and Device Discovery: Protocols and Programming, McGraw-Hill, 2002.

[2] Guttman E., et al.: Service Location Protocol, Version 2, IETF Request for Comments 2608, June 1999. Available at: http://www.ietf.org/rfc/rfc2608.txt

[3] Ferscha A, et al.: Digital Aura, Advances in Pervasive Computing, part of the Second International Conference on Pervasive Computing (Pervasive 2004), Austrian Computer Society (OCG), Vol. 176, pages: pp. 405-410, April 2004

[4] Jonvik Tore E., et al.: Building a Virtual Device on Personal Area Network, Proceedings of 2003 International Conference on Software, Telecommunications and Computer Networks (SoftCOM'2003), Dubrovnic (Croatia) / Ancona, Venice (Italy), October 7-10, 2003.

[5] Niemegeers I. and Hememstra de Groot S.: Personal Distributed Environments for Mobile Users, Mobile Europe 2003, European conference on mobile solutions, 2003. Available at http://katanga.bbn.de/mobile_europe_2003/

[6] Arbanowski, S., et al.: I-centric communications: personalization, ambient awareness, and adaptability for future mobile services, Communications Magazine, IEEE, Volume: 42 , Issue: 9 , Sept. 2004.

[7] Atkinson, R. C., et al.: The Personal Distributed Environment, Wireless Personal Multimedia Communications 2004, September 2004.

[8] Porras, J., et al.: Peer-to-peer communication approach for mobile environment, 37th IEEE Annual Hawaii International Conference on System Sciences (HICSS), 2004.

[9] Jäppinen, P., et al.: ME: Mobile E-Personality, WSEAS Transactions on Computers, Volume 2, Issue 2, April 2003.

[10] Morrow, R.: Bluetooth operation and use, McGraw-Hill 2002.

[11] Zhang X. et al.: Bluetooth Simulations for Wireless Sensor Networks Using GTNetS, The IEEE Computer Society’s 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'04), 2004.