Implementation of a cellular framework for Spontaneous Network Establishment

Marc Danzeisen[(1)(2)], Torsten Braun, Simon Winiker
Computer Networks and Distributed Systems[(1)]
University of Bern
Bern, Switzerland

Daniel Rodellar
Innovations[(2)]
Swisscom AG
Bern, Switzerland

Abstract— **Wireless communication technologies enabled the possibility of building spontaneous networks between two or more users to exchange data. The problem in the establishment of such networks lies in the configuration that has to be agreed on and in the way the communicating parties can be identified. In prior publications we have presented our vision of convenient networking in a heterogeneous environment. In this paper we describe an implementation that offers a dashboard-like tool, which can, with the help of a cellular network, ease the formation of spontaneous networks among heterogeneous nodes. Furthermore the provided implementation is able to secure the acquired communication links in the spontaneous network and therefore protect the exchanged information against possible abuse.**

Spontaneous Networking, WPAN, Cellular, wireless LAN, Bluetooth

I. INTRODUCTION

Many research efforts in the domain of spontaneous networking are aiming at providing means to enable devices to communicate with little or no user knowledge about the underlying technology and its configuration. The establishment of communication channels should happen in an ad-hoc and convenient manner for the user. It also should be possible to connect at any place, at any time and with anyone who is interested in communicating. In the envisioned scenarios we do not limit ourselves to Mobile Ad-hoc Networks like defined in the IETF workgroup MANETs [4], even if some principles might appear in the provided implementation or influence our future work. In contrast to pure ad-hoc networking, the presence of infrastructure is fairly accepted and up to a certain degree even desired in our considerations. To clarify the scope of this paper, the following example scenarios are provided. Note, that these scenarios are only examples, and do not limit the general scope of our work. Alice and Bob represent business users, who do not have deep knowledge about computer networking:

Scenario 1: Alice and Bob want to securely interconnect their devices, which are both wireless LAN (WLAN) enabled.

Scenario 2: Alice and Bob want to securely interconnect their devices, which are both Bluetooth enabled.

These two scenarios have been used to prove the basic correctness of our concept called Cellular Assisted Heterogeneous Networking, which provides a framework to offer robust and user-friendly connectivity establishment between two or more nodes in an ad-hoc manner, using various communication networks and therefore heterogeneous communication channels.

In contrast to related work like SyncTap [17], where the devices involved within the spontaneous network belong to the same administrative authority, we can not assume synchronized operation of the different participating nodes in our envisioned scenarios. Hence, Alice and Bob do principally not trust each other.

However, these two chosen scenarios contain the main challenges in establishing protected connections between communication devices in a convenient way. The extension of our implementation to cope with other communication technologies than WLAN and Bluetooth is ongoing work. More information about our general vision of convenient heterogeneous networking including also infrastructure based communication channels can be found in [1][2][3].

The rest of the paper is structured as follows: In section 1 we first identify the issues that have to be solved, and the challenges to be faced, when providing spontaneous networking functions. Then, a possible solution is introduced to deal with the previously identified issues. Defining the minimum required message exchange between the involved entities to successfully set up a communication channel, the implementation architecture is derived. A message exchange protocol is defined and described in section 5. Section 6 and 7 explain in detail how WLAN and Bluetooth links can securely be established using the implemented protocol. The evaluation of the implementation is presented in section 8. The last chapter concludes and gives an outlook on future work.

II. ISSUES AND CHALLENGES OF PROVISIONING CONVENIENT SPONTANEOUS NETWORKING

For a successful deployment of spontaneous networking certain issues have to be solved:

To establish a communication channel, the nodes have to be reachable and addressable in available networks with public addresses, i.e. in the Internet with public IP addresses.

As such addresses are not always static they do not always identify the same device. Furthermore, mobility often forces

the nodes to change the IP addresses. Even when Mobile IP [5] is deployed, and the home addresses are statically assigned, the nodes are only reachable when attached to an IP network and registered with the home agent. When taking into account that most public IP access networks are charged based on time, being always connected is not yet really attractive. Therefore, well known identifiers are needed to tackle a communication peer. In contrast to communication addresses, these identifiers must be static and always available at a reasonable price, while the communication address may change.

For that reason, a lookup service has to be provided, to resolve a given identifier to the currently used communication address(es) (e.g. the Mobile IP home address). The most known service offering such address resolution is the Domain Name Service (DNS), which delivers IP addresses belonging to a given Fully Qualified Domain Name (FQDN) [6]. The fact that IP addresses are often allocated in a dynamic and temporary way makes address resolution services like DNS vulnerable to integrity problems. Nodes have to regularly update the DNS about the actually allocated IP addresses. Using other identifiers as primary identifiers, where the identifier/address binding is less dynamic helps to avoid this integrity problem. In the next section we will describe the advantages of reusing the cellular Mobile Subscriber Integrated Services Digital network Number (MSISDN)/International Mobile Subscriber Identity (IMSI) binding as the primary identifier/address pair for communication establishment.

After having resolved the correct communication address from the primary identifier, it is possible to establish a connection between two devices. In order to protect the communication, this channel has to be encrypted. To provide encryption, the communication parties have to agree on several security parameters, like keys or encryption algorithms. To ensure the authenticity of the originator and receiver of these parameters and to prevent attacks like the man in the middle attack the identifier/address bindings have to be authenticated. Furthermore means have to be offered to locate users in dynamic environments, like mobile networks.

III. USING THE CELLULAR NETWORK AS ENABLER FOR SPONTANEOUS NETWORKING

As mentioned before we suggest reusing the cellular network to cope with the issues identified in the previous section:

The most obvious advantage of this approach lays in the static identifier offered for each user, the MSISDN. Another big advantage of this approach is the independence from a specific addressing scheme. Implementations with Bluetooth addressing or IP addressing were made and proved the benefit of that flexibility. Details on these implementations are presented in section 6 and 7.

For the translation of the identifier (MSISDN) into specific communication addresses, we chose an indirect approach, where the resolution is done in the end nodes. The cellular network of choice for our implementation was the GSM network, even if the basic principles purposed can also be applied to other cellular networks.

The GSM network provides a secured and reliable communication channel for the exchange of the needed setup messages between the involved users. The mechanism, that we used in our implementation to exchange messages over the GSM network was the Short Message Service (SMS) [7].

This secured channel can also be used to exchange the needed security parameters between the users, so that the future communication channel between them can be encrypted and thus protected. The needed security parameters depend on the specific technology, which is used for the communication channel. In the case of Bluetooth this is a PIN code, for example, and in the case of WLAN a Wired Equivalent Privacy (WEP) key.

The needed authentication is provided implicitly by the cellular operator. In case of the GSM network this is achieved using the Subscriber Identity Module (SIM) [8]. Finally, GSM paging [9] allows discovering users, so that setup messages can reach their destination (based on the resolved IMSI belonging to the desired MSISDN). Fig. 1 intends to give a summary of the used GSM principles within our implementation. The numbered list following the figure explains the single steps and summarizes the advantages of using a cellular network to setup spontaneous network links.

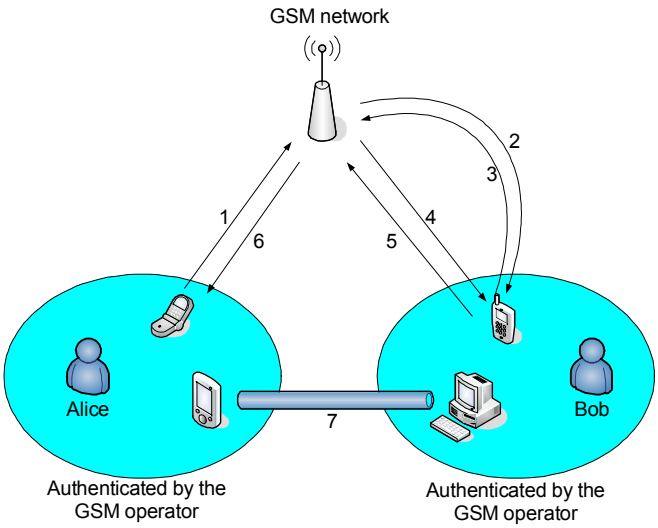

Figure 1. Spontaneous network link establishment with help of a cellular network

1) Connection request from Alice to Bob's MSISDN, sent via SMS. The request includes the communication address (i.e. IP address) of Alice's PDA

2) GSM paging to locate Bob's GSM device (after having resolved the IMSI belonging to Bob's MSISDN)

3) GSM paging response

4) Delivery of the connection request from Alice via SMS

5) Connection response including the communication address (i.e. IP address) of Bob's computer and the connection and security parameters to Alice's MSISDN via SMS

6) Connection response of Bob via SMS

7) Secured link establishment between Alice's PDA and Bob's computer

Note that the cellular network offers the required transport channels (for instance SMS) for the needed information exchange, including authenticated identifier/address resolution and paging mechanisms. The next section focuses on the required information exchange and the structure of the different messages.

IV. BASIC MESSAGE EXCHANGE

By using a cellular network to exchange setup messages and security parameters one can cope with the most important issues of spontaneous network formation. The goal of this chapter is to define the minimal set of messages needed to successfully complete such a connection establishment.

It is quite obvious, that at least a connection request is needed, in which Alice can announce to Bob her intention to communicate with one of his devices. This request can already include required communication parameters. In return a connection response is needed for Bob to indicate his acceptance of the request and to indicate his parameters to Alice. With these two messages, it is already possible to enable a spontaneous network link establishment. Additionally, an error message has to be present, in case unforeseeable events in the network prevent Alice and Bob from the establishment of a spontaneous communication channel. Having these three messages (communication request, communication response, error message) in mind we defined a simple implementation architecture to realize the protocol.

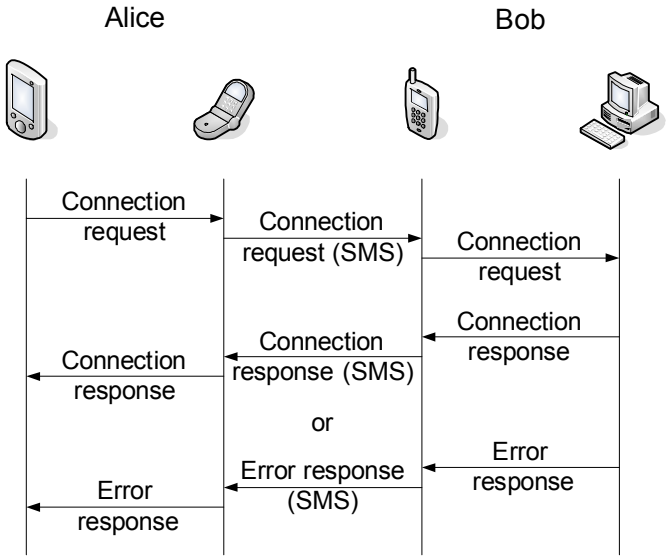

Figure 2. Messages exchange

V. IMPLEMENATION ARCHITECTURE

As it can be seen from the diagram in Fig. 2 the messages are delivered over the GSM network, and need to be converted in a suitable format for their transport over the GSM network. We decided to use SMS, because this service is widely available and no further interaction is necessary with the network operator to get our protocol messages delivered via the GSM network. Again, other mechanisms than SMS exist and therefore, the implementation must be easily expandable to cope with these other technologies.

To use the SMS service the protocol messages have to be converted in SMS compliant messages. For sake of simplicity, an ordinary cell phone was used as the interface to the GSM network, and controlled over a serial connection with help of AT commands. As these phones have limited functionality, the message conversion is realized within the laptop. But as there are more and more smart phones available, which include the necessary capabilities and programming interfaces it can be imagined, that the implementation can also be realized on the GSM device rather than on the laptop.

To allow future migration the architecture must be flexible enough to be adopted easily. For that reason we decided, to isolate the conversion function from the main application in order to make the implementation independent of the underlying message transport system. This isolated component is called adapter. For each cellular message delivery mechanism (e.g. SMS, USSD), that can be used, a separate adapter can be developed with regard to its characteristics.

The same problem applies also to the interaction with the communication technology that is used to establish a data connection. In the defined scenarios, it is possible that the devices can have several communication technologies available like Bluetooth and WLAN. Therefore, also this part was isolated and the resulting component is called connector. This component is responsible to apply the parameters agreed on during the communication setup negotiations to the respective network interface card (NIC) and to handle related requests and responses. It is the responsibility of the main logic to choose the connector in charge for the current communication technology.

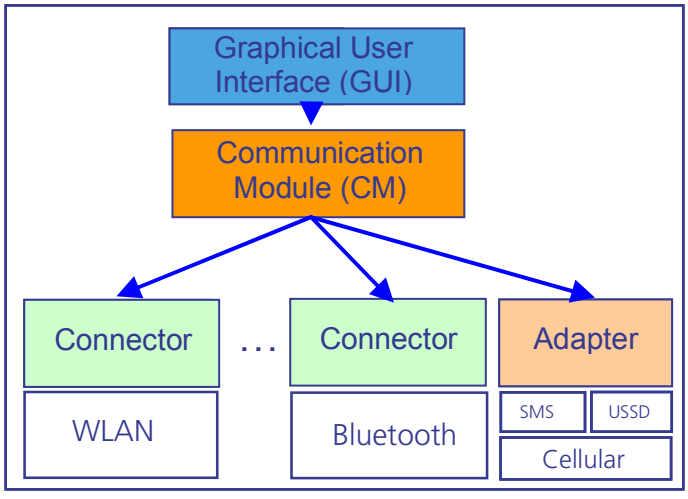

Figure 3. The implementation architecture

This main logic is implemented in a component called Communication Module (CM) and is mainly responsible for the management of the different messages. The CM relays

messages to the related component, i.e. to the adapter, if the messages have to be sent over the GSM network, or to the connector, if the messages have to be handled locally.

The CM offers a standard socket interface to the adapter, which can also be used by a user interface. For our purpose, we decided to implement a graphical user interface (GUI) to enable convenient spontaneous networking. With help of this GUI, the user can invoke connection requests and configure his application. Fig. 3 shows the schematic structure of the application that was implemented. More details about the overall architectural design can be found in [1].

With this implementation design we assured the necessary flexibility to adopt the implementation to support additional GSM message delivery mechanisms and also future communication technologies. The next section describes the detailed protocol message structure.

VI. THE PROTOCOL

The design of the implementation architecture influences the formatting of the needed protocol messages. Since a message has to pass through several components, different kinds of information are needed. E.g. the CM must know where the message comes from, and to which component it has to relay the message to. The adapter has to know, to which MSISDN it has to send the message to, and the connector needs the negotiated parameters to configure the NIC accordingly.

Therefore, the message structure consists of three major parts. The first part is the header, which contains basic information relevant for the CM. The second part is the cellular header, which defines the cellular network specific information, which depends on the used adapters. The third part contains so-called service specific parameters, where the service, depends on the used communication technologies and is important for the connector. Fig. 4 illustrates the message format, which can be applied to all of the three basic message types; the connection request, the connection response and the error response. A typical protocol message has a length of 57 Bytes (connection request) and 67 Bytes (connection response) and therefore fits into a single SMS message.

The following list briefly explains the different fields defined in the protocol messages.

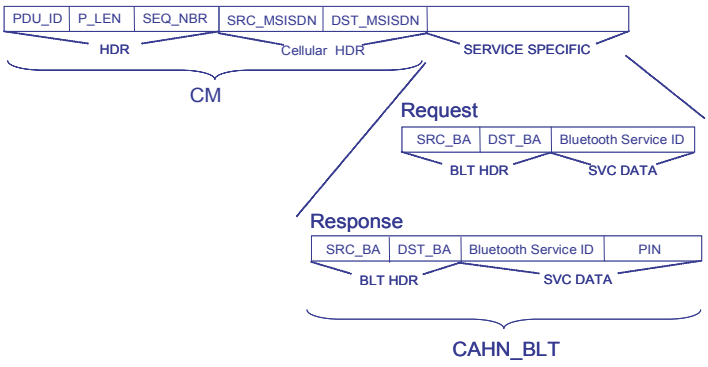

Figure 4. The Protocol

Header:

- PDU_ID contains a unique ID for the identification of the message type, i.e. 0x01 for a connection request.
- P_LEN indicates the length of a network package.
- SEQ_NBR is needed for reassembling fragmented packets, which may occur because of the limited length of SMS messages.

Cellular header:

- SRC_MSISDN is the identifier of the message sender.
- DST_MSISDN identifies the messages receiver.

In the service specific part, the content of the messages can vary according to the used NIC. For a Bluetooth related response, for example, it can contain the following fields:

- SRC_BA indicates the Bluetooth address of the messages originator. Based on which the indirect lookup is done, and the relation between the identifier (MSISDN) and the connection address (Bluetooth Address (BA)) can be established.
- DST_BA shows the Bluetooth address of the message receiver.
- PIN contains the PIN code needed to establish a secured Bluetooth connection.

VII. CELLULAR ASSISTED WLAN CONNECTION ESTABLISHEMENT

The first scenario that was implemented enables spontaneous network establishment between two nodes equipped with WLAN in ad-hoc mode. For the exchange of configuration and security parameters, the GSM network has been chosen, i.e. SMS for the transport of our protocol messages. The cellular header of the protocol PDUs for the use with GSM SMS is shown in Fig. 4.

In this specific scenario, the configuration information consists of the ESSID of the formed WLAN network and the common WEP key. With these parameters, the WLAN connection can be established securely. To carry this information, the service specific part of the protocol messages was formed according to the protocol definition described in the previous section. In the implemented application a user can prepare his device to support the spontaneous network formation, by setting up a fixed default ESSID. Upon a connection request this ESSID is used to form an ad-hoc WLAN and is included in the connection response message. The WEP key used to secure the communication is either computed randomly or entered manually by the user.

A user intending to establish a spontaneous WLAN link can send a connection request to the respective user over the GSM network (using the MSISDN as primary identifier allowing even the usage of the phone book). This request is then handled on the device of the receiver, where the WLAN interface is

configured and a connection response is returned containing the used service specific parameters (ESSID and WEP key) to the initiator, again via the GSM network. Finally, the requesting node configures its WLAN interface according to the received service specific parameters. After these steps both devices are configured and the connection between them is established.

This scenario was implemented in a test bed and proved to work in the expected way. Nevertheless, there is still an important assumption in the scenario definition. The application works only if the two devices, intended to communicate are in the vicinity of their WLAN radios. If a communication is not possible because of missing coverage, the setup messages are exchanged anyway, even if it does not result in a successful communication channel establishment. Therefore, means have to be provided to test whether a peer is within communication range, so that the parameter exchange is not done without actual successful connection establishment. The main problem can be reduced to the missing ability to scan the environment for a specific primary identifier like the MSISDN. The only information that can be retrieved by a scan is the communication address, i.e. the MAC address of the WLAN nodes within the neighborhood. Hence, reverse lookup is required to resolve primary identifiers, which uniquely identify communication parties.

Bluetooth offers better means to scan for primary identifiers. In the following section we describe, how we managed to use Bluetooth Service Discovery mechanisms to do so.

VIII. Cellular assisted bluetooth connection establishement

According to the Bluetooth specifications [10], Bluetooth equipped devices support the Bluetooth Service Discovery Protocol (Bluetooth SDP) [11]. This protocol offers the possibility to announce service specific information to other Bluetooth devices in the vicinity of the Bluetooth radio. With help of this feature, it is possible to cope with the issue shown in the WLAN scenario.

To provide a solution, a new Bluetooth service has been defined to announce the MSISDN of the device's owner to other Bluetooth devices in range. Nodes can thus scan the environment for desired peers before starting further negotiations.

Services for Bluetooth are defined with the help of so-called profiles. Within these profiles, services can be described by the help of service attributes. For our purpose, we defined a new profile, describing our spontaneous networking service, with the MSISDN as a service attribute. Practically this forced us to change the source code of the Bluetooth stack to add the new service.

To guarantee the interoperability with existing Bluetooth services, we decided to define also a service attribute to indicate the capability of supporting our application. This service attribute represents all Bluetooth services (i.e. Personal Area Network (PAN) profile [12]) that can be accessed based on our spontaneous networking application. Since a simple Bluetooth connection between two nodes is based on such a service (or profile), this introduced service attribute allows an easier integration of further Bluetooth connection types (i.e. Generic Object Exchange profile [13] a.o.).

Using the GUI of our implementation, the user can easily prepare his device for spontaneous networking by enabling this defined spontaneous networking service (implemented as a Bluetooth profile) and define which of his Bluetooth services (i.e. PAN a.o.) support the spontaneous networking establishment.

On the other hand, a user who wants to establish a spontaneous network link uses the GUI to easily browse his environment for the services belonging to a specific MSISDN and thus a specific user. If he can find the desired service, he can invoke the connection request over the GSM network. To avoid the transmission of a connection response to users, which are not physically reachable, the Bluetooth inquiry mechanism may also be used before. Therefore, the application would scan for the Bluetooth device, which's Bluetooth device address (BA) was contained in the connection request. If this address can be found, a connection is physically possible, and a response makes sense.

With the definition of these additional service attributes and scans, it is possible to cope with the issue presented in the WLAN scenario description. To realize these concepts, the protocol messages were defined to carry the Bluetooth related parameters. These are the source BA, the destination BA, the identifier of the requested Bluetooth service and the PIN code for the security (remember Fig. 4). Further a connector was added to handle the Bluetooth devices and services.

The implementation showed, that messages over the GSM network were only exchanged, when the connection was physically possible.

IX. Evaluation

The first basic measurements made with the application running on a test-bed aimed to prove the usability and to identify improvement potential.

The total establishment time is of major interest in order to guarantee a maximum user benefit. Therefore, we measured the time for an entire Bluetooth connection establishment. Time measurement starts, when the connection establishment is invoked on the client and stops as soon as the connection is established. 20 measurements were made and Fig. 5 shows the results of these tests: the times measured for a Bluetooth connection establishment with our application ranged from 32.7 sec in minimum up to 46.2 sec in maximum. To explain the high values and the high variance among further measurement were done.

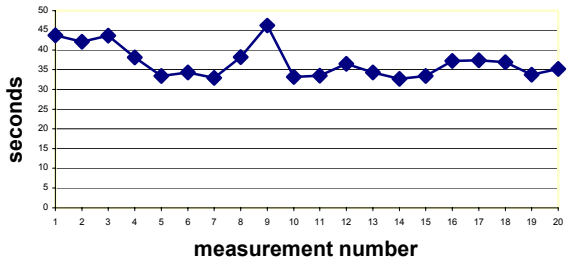

Figure 5. Connection establishment times

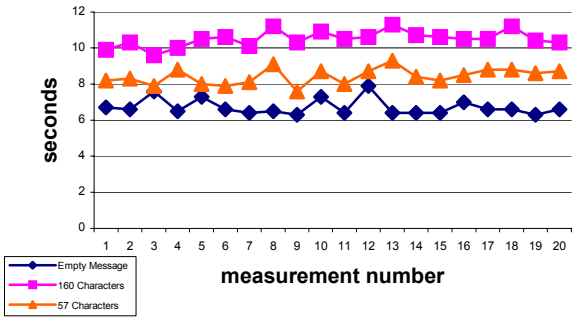

Figure 6. SMS message exchange delays

The most obvious parameter influencing the total time is the SMS exchange delay. Fig. 6 shows the measurements for messages with a different length.

As can be seen in the presented chart, the SMS message exchange delays depend on the size of the message (for our application the connection request message for Bluetooth contains 57 characters). The average value for the 57 characters long message is 8.4 sec.

Further on, the Bluetooth connection establishment time has to be analyzed. This is the time the Bluetooth stack takes to establish a service-specific link. In our case, this was the connection establishment using the Bluetooth PAN profile. Again, 20 measurements were taken and the results can be seen in Fig. 7.

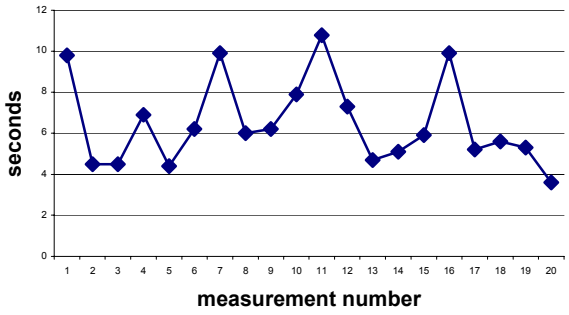

Figure 7. Bluetooth PAN connection establishment times

The first impression we can gain from this chart is the high variance, which can explain the high variance of the entire connection establishment process. The average value for a Bluetooth connection establishment using the PAN profile is 6.5 sec. This value is caused by the Bluetooth protocol stack and not directly by our implementation. Analysis of the connection establishment time of WLAN interfaces resulted in values below 1 sec, which is not surprising, when considering the absence of any time consuming handshake process in WLAN.

The third parameter having a remarkable impact is the time a Bluetooth inquiry takes. This value can directly be influenced and ranges from 0 sec up to 10 sec, depending on the required fidelity of the inquiry. In the measurements taken for the entire connection establishment process, this inquiry value was set to 10 sec to offer the most reliable result.

The three delays (SMS message exchange, Bluetooth inquiry and connection setup) together explain the total establishment time. We can not influence the Bluetooth PAN connection establishment time and the inquiry time is a trade-off between reliability and delay. This accuracy of the neighborhood detection has a direct consequence on the ability of the system to avoid unnecessary communication setup negotiations, mentioned in section 6 & 7. Hence, the parameter we will focus on for future enhancements is clearly the SMS message exchange delay. For that purpose, further investigations have been started to replace the SMS mechanism with a more suitable approach, like the Unstructured Supplementary Service Data (USSD) [14] offered by 2G/3G networks.

X. CONCLUSION

Our main objective was to enable secured spontaneous networking in a user friendly way. For the initial configuration and security parameter exchange we chose to make use of a cellular network, which helped us to cope with the major issues of spontaneous networking. In a second step the implementation architecture and a basic protocol have been defined, as the basis for a demonstrator application, which was intended to verify the concept and to provide us with important empirical knowledge. This application proved the value of our approach and showed that it can serve as an enabler for user friendly spontaneous networking.

Within the WLAN scenario a serious issue has been identified: Neighborhood detection mechanisms offered by the WLAN standard are limited to communication addresses (i.e. MAC). The dynamic and temporal acquirement of communication addresses (i.e. different devices used by the same person) makes it very difficult to correctly correlate such communication addresses to communication peers (i.e. persons). In other words, having scanned a specific WLAN MAC address does not resolve the primary identifier of the communication peers using that WLAN device. Hence, it is not possible to know, whether a user to which a connection shall be established is actually in the physical vicinity of the radio. To provide a solution for that problem research focuses on the usage of Bloom filters [15][16] to detect whether a certain node (identified by his MSISDN) is reachable.

To prove our concept, we defined a Bluetooth scenario, where we presented a mechanism to resolve communication addresses to primary identifiers using service discovery

attributes. The implementation of this scenario was the basis for the evaluation at the end of this paper.

This evaluation helped to get an impression of the different factors influencing the spontaneous networking connection establishment. Wherein, the usage of SMS to exchange our protocol messages introduced most of the delays experienced in our implementation. To improve this aspect, work will go on in the area of USSD, which could offer us an alternative channel over the GSM network. The connection oriented characteristic of a USSD channel should suit better the requirements of our application, but requires closer collaboration with mobile network operators.

The work done and presented in this paper brought us closer to our goal of a convenient spontaneous heterogeneous networking. However, it also revealed the complexity of practically integrating different technologies to offer a user friendly communication platform. Especially, the connection maintenance when the participating nodes are changing communication technology (i.e. due to movements, or changing bandwidth needs), becomes a very difficult task. The existing implementation restarts the connection setup whenever the data link is lost. In future work a more pro-active behavior is considered, where alternative data channels are managed continuously throughout the whole communication session allowing a dynamic handover if needed. Furthermore, inbound signaling over an established data link will improve the performance of the system. In that case the usage of the cellular would be limited to the bootstrapping phase and as fallback signaling channel.

[1] M. Danzeisen, R. Rodellar, T. Braun, S. Winiker, “Heterogeneous networking establishment assisted by cellular operators”, The Fifth IFIP TC6 international conference on mobile wireless communications (MWCN 2003), Singapore. October 2003

[2] M. Danzeisen, R. Rodellar, S. Winiker, T. Braun, ”Heterogeneous Networking facilitated by Cellular Networks”, Comtec Magazine 03/04, June 2004Mobile

[3] S. Winiker, ”Integration of Cellular Assisted Heterogeneous Networking and Bluetooth Service Discovery Protocol”, Diploma thesis, University of Berne, May 2004

[4] Ad-Hoc Networks (manet), working group

[5] C. Perkins, “IP Mobility Support for IPv4”, RFC 3220. January 2002

[6] Zaw-Sing Su, Jon Postel, “The domain naming convention for internet user applications”, RFC 819. August 1982.

[7] ETSI standard, “Point-to-Point (PP) Short Message Service (SMS) Support on Mobile Radio Interface”, GSM 04.11

[8] ETSI standard, “Subscriber Identity Module (SIM) functional characteristics”, GSM 02.17

[9] ETSI standard. “Basic call handling”, GSM 03.18

[10] Bluetooth SIG, “Specification of the Bluetooth System, Version 1.2”, November 2003

[11] Bluetooth SIG, “Service Discovery Application Profile”, November 2003

[12] Bluetooth SIG, “Personal Area Network Profile”, November 2003

[13] Bluetooth SIG, “Generic Object Exchange Profile”, November 2003

[14] ETSI standard, “Unstructured Supplementary Service Data (USSD)”, GSM 02.90

[15] Burton Bloom, “Space/time trade-offs in hash coding with allowable errors”, Communications of ACM, pages 13(7):422-426, July 1970.

[16] E. Maghsoudi, “Design and Implementation of WLAN Support for Cellular Assisted Heterogeneous Networking”, Diploma thesis, University of Berne, November 2004

[17] “SyncTap: synchronous user operation for spontaneous network connections”, ACM Personal and Ubiquitous Computing, vol. 8, no. 2,May 2004