ConCom – A language and Protocol for Communication of Context

Albert Krohn, Michael Beigl, Christian Decker, Philip Robison and Tobias Zimmer

Telecooperation Office, University of Karlsruhe
{krohn,beigl,cdecker,philip,zimmer}@teco.edu
http://www.teco.edu

Abstract. With ConCom, we address the area of communication in wireless networks. We focus on heterogeneous settings with highly mobile devices carrying limited resources. ConCom present a way to express and communicate information, especially context, in a way that is similar to a naturally spoken language. ConCom uses sentences with a subject and attributes in its structure to represent and organize the transport of context and data. ConCom works connectionless and without addressing and forms an efficient way to exchange information in ubiquitous computing environments. We implemented ConCom and show applications taking advantage of it.

1 Introduction

Progress in the area of ubiquitous computing platforms and sensor networks has undergone acceleration in recent years. More and more systems are being developed and are being brought to the product state. Some of the popular examples are the Motes [2], the Eyes platform [3], and the Smart-Its [5], based on different hardware and communications technologies including infrared [1] or wireless radio communication. In Ubicomp settings many highly mobile, unsupervised, infrequently administered and manifold computing devices spontaneously have to work together. This fact raises typical problems surrounding the organization and communication of data and context. The cooperation of devices is in most cases based on services, which have to be discovered and matched before applications can take advantage of the services found. The established service discovery mechanisms, used in approaches like Jini [6] or UPnP/SSDP[19], are exhaustive and the subsequent, post-discovery communication is based on a static binding between communication partners. In fact this is the case in standard wireless data exchange protocols such as WLAN (TCP/IP) or Bluetooth, which are based on a connection between explicitly addressable network instances. The addresses used to build up the connections have to be known a priori or are “looked-up” in an available registry service that supports the discovery process. Therefore, establishing and maintaining connections in highly mobile and heterogeneous environments is complicated and produces a lot of overhead. This overhead consumes power and bandwidth, which are both limited resources on small computing nodes often used in UbiComp applications.

With ConCom, we found an efficient way to exchange context information and data between instances of a network based on the content of the transported information and not based on addresses or bindings between addresses. This followed our repeated observation in experiments carried out over several years, where the information sent from one device in an Ubicomp setting tends to be of interest and consumable by more than one specific device or group. ConCom supports this observation by offering an easy way to organize broadcast and multicast transmission of context and data. The data to be communicated is sent out into its physical range, where devices with interest in the contents may reside. Filtering of information is then done at the consumers' side based on semantic content filtering.

By way of analogy, an everyday application that is based on broadcast and content filtering without addressing is television and radio. The different channels of radio and television programs are broadcasted to all end devices like TV sets or radio receivers, over various frequencies, but the end user selects programs of interests by tuning in to a particular channel and program based on an assessment of the content. ConCom works in a similar way in that information is generally broadcasted and the consumers filter out their content of interest.

This paper introduces ConCom by first presenting the structure of the language for data and context representation. This data structure is the core element of ConCom and precondition for the communication protocol based on ConCom, which is described in section 3. Associated issues like location-based delivery, routing and security are also discussed. Section 4 shows the strength and application of ConCom in practical examples of Ubicomp applications.

2 Structure of the ConCom Language

ConCom is a way to represent and exchange context, especially designed for ubiquitous and pervasive computing as well as sensor networks. It considers the issues of energy saving, efficient broad- and multicast as well as content filtering and supports sensor fusion already in the data representation. Nevertheless, the type of context to be exchanged through ConCom is not limited to the area of sensor systems. It can transport any type of data and can be implemented on basically any existing network that supports broadcast. In ConCom, context or data is represented in a manner comparable to naturally spoken language. The organization of data traffic is not based on end-to-end connections, but is based on the use of sentences to transport information between communication partners. The ConCom sentence is the basic element of the communication language. If an instance of the ConCom network wants to transmit data, a sentence has to be built, that follows the typical structure of this sentence specification.

How does a sentence look in ConCom?

A sentence in ConCom always begins with a subject. This subject is a logical identifier for the originator of the transmitted data. In our settings subjects normally refer to devices such as thermometers or PDAs or simpler everyday objects like pens and coffee cups. In other domains, subjects can also be virtual references including network management functions or groups of devices or even software programs. Influenced by the flexibility and variable abstractness of a spoken language, a subject is an identifier of an individual, class or role. In a conversation, we identify persons by their individual names or group them together in a team and collectively address them. One example is the management hierarchy in large organizations where individuals are grouped in teams, then in departments, divisions and so on with increasing abstraction, each level possessing its own identifier/ descriptor. Translated to the world of sensor systems, a subject could be an individual such as "`the thermometer with serial number 3374629`" or an aggregation of individuals - "`TEMP2000 thermometers`" or any other abstract class to represent a subject.

In the ConCom sentence, the subject is followed by an arbitrary number of *attributes* – separated by commas – that all correspond to the subject. These attributes represent context information. We understand "context" here as the properties of the subject or actions taking place with which the subject involved. The number of attributes is generally not limited but all attributes have to be and are assumed related to the subject. An attribute descriptor is stated *with* or *without* an additional attribute value i.e. the value and descriptor are synonymous. All ConCom sentences are terminated with a full stop, allowing the beginning of a new sentence. Although the ConCom sentence is primarily designed to transport context data, any application data can be embedded and understood as attributes or actions of objects and hence be embedded in the ConCom sentence frame.

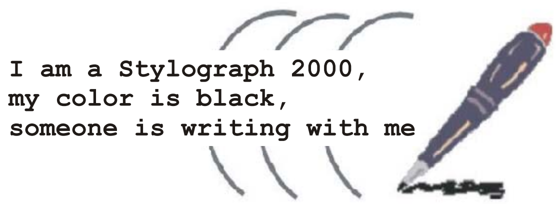

Figure 1: context data broadcasted from a Pen to the environment

To explain the construction and consumption of these sentences, figure 1 depicts a simple example of a communication in the AwareOffice [13]. The pen (Stylograph 2000) communicates information about itself into the environment. The sentence would e.g. consist of the following parts:

- Subject: Stylograph 2000
- Attribute: color (attribute descriptor): black (attribute value)
- Attribute: someone is writing

These are then combined in the sentence: "`I am a Stylograph 2000, my color is black, someone is writing with me`." Another example in a sensor network is a wireless thermometer. It could broadcast its information like this: "`I am the TEMP2000_3374629, my temperature is 25 degree Celsius.`" While the subject in the first example referred to a class of pens (Stylograph 2000), the subject in this case represents a specific thermometer device. Other examples, e.g. transmitting data like files would look like: "`I am the PDA of Albert, I transmit data fragment number 63 of file con.hex, its content is:00110001001… .`"

These examples show that building sentences is possible with all kinds of required data transmission, given that data is explicitly assigned to a subject.

3 Communication based on ConCom

The ConCom sentence structure helps the instances of the network to filter the broadcasted messages. The communication is generally based on a *broadcast-and-subscribe* mechanism and is connectionless.

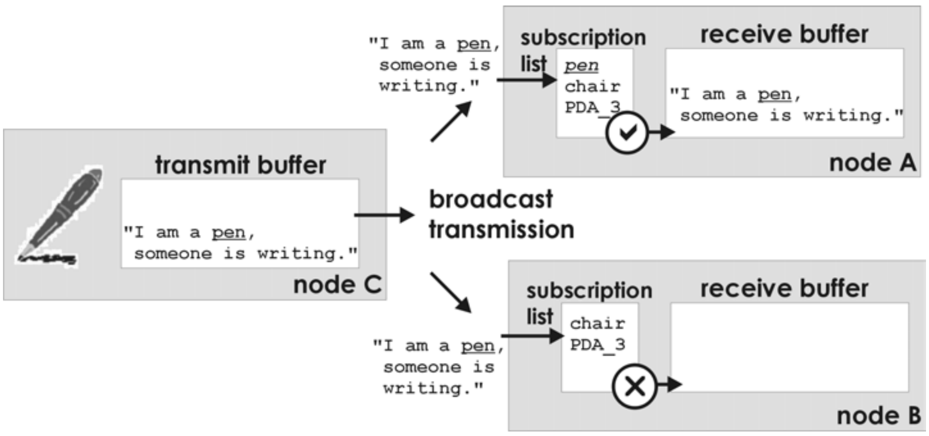

Figure 2: content filtering on consumers' side

All messages in the ConCom protocol are broadcasted to all available network instances in the target cell(s). In order for each receiver instance to filter relevant data, each maintains a local subscription list containing its "subjects of interest". All traffic on the network is received by each network instance and the sentences are filtered based on its subscriptions. That is, only sentences with subjects contained in the subscription list are kept, while the others are discarded. Subscriptions within each instance are not static and can be changed and renewed at any time. Figure 2 shows a situation where the network instance of the pen device (named as node C) broadcasts the message "`I am a pen, someone is writing`". The message arrives at two network instances that run in sensor nodes, node A and node B. Every network instance holds its own subscription list locally. The two nodes have different subscription lists – node A is interested in the subjects pen, chair and PDA_3, whereas

node B is only interested in the chair and PDA_3. The broadcast message from the pen arrives at both nodes as they are in the broadcast area of the pen. Node A has the subject “pen” in its subscription list, such that the message is accepted and placed in the receive buffer. Node B doesn’t carry a subscription to the “pen” in its subscription list thus the message is not received but discarded in an early state.
Using this broadcast-and-subscribe mechanism, each instance only accepts and processes data from the subjects that exists in its subscription list. This is the core functionality of ConCom. Conversely, in traditional networks like TCP/IP or Bluetooth all traffic is destination oriented. Packets are routed or accepted based on their destination address and service ports or types. The destination address is used to filter messages, to update routing tables, to establish connections, and basically to organize the entire network traffic. However, ConCom inverts this methodology by minimizing the relevance of a destination address and specifying an originator-based filtration carried out at the receiver. The destination is no longer an explicit or virtual reference but is simply the physical region and range within which the data is broadcasted.

3.1 Possible Addressing in ConCom

Although the focus of the paper has been on the benefits of destination address free communication, there are possibilities to introduce addressing into the ConCom language where the application sender of the network instances requires such e.g. performance of routing. The way we generally do this is to define addresses and routing information as attributes of a subject and hence include them in the sentence. All instances that support routing would have to subscribe to all subjects, parse the incoming sentences for routing information and forward the packets accordingly. From there on, all known routing techniques can be used. The consequence of this sentence encapsulated addressing style is, that a router would have to parse a packet to look for routing information, which is very inefficient. A more appropriate and supporting style of addressing and routing for ConCom is found in the RAUM system:

3.2 RAUM in ConCom

The RAUM system [7] for location-based communication in interactive environments provides the sender of a sentence with the ability to address a certain spatial area. Messages that are addressed to a spatial area are only broadcasted in network cells that lie within that physical space. The spatial addressing is organized in a tree structure encompassing semantic and geographic location descriptions. Figure 3 shows an example of a location tree for an organization. The leaves of this tree – representing rooms in this example – can be extended by geometric coordinates relative to the local point of origin of that leave. In this way communication can easily be addressed to a certain location as described in [7] in detail. Receivers of RAUM communication carry an internal list of locations they want to receive data from. In this aspect RAUM works similar to ConCom and is a consequent evolution from the broadcast and subscribe mechanism to support semantic location routing.

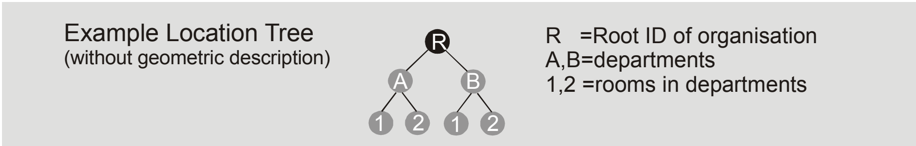

Figure 3: RAUM location tree

By using the RAUM system in conjunction with the ConCom language the semantic source filtering process is extended to two layers. On the lower layer the RAUM filtering based on locations of interest is performed whereas, on the ConCom layer, filtering by content is performed as described earlier.

The basic setting of a ConCom network incorporates only one wireless communication cell as the target of broadcast. As a generalization, RAUM coordinates communication when several cells are combined. Routing data among these cells can be performed using the location based routing techniques of the RAUM system. The RAUM routing system forwards data packets to locations of interest where they can be locally broadcasted again. Every router in this system is responsible for a semantic location. In figure 3 a router would be located in one of the nodes and be responsible for this node and all subordinated ones. The router is aware of all these locations. If it receives a packet not intended for a location it is responsible for, the packet is forwarded to the next higher router in the location tree. Analogous to the root servers of a DNS system in the worse case a packet has to be handed up to the router assigned to the location trees root. Whenever a router receives a packet for a location it is responsible for it forwards the packet to the network partition associated with that location.

3.3 ConCom as an Inter-Protocol Communication

ConCom has a very simple structure and requires no special features of underlying network protocols. ConCom can be implemented on top of any existing network protocol and work as a mediator between network instances in applications that are distributed among devices with different network protocols and media. The only obvious necessary technical pre-conditions are that the networks have to be interconnected via e.g. bridges and gateways that transfer the actual payload between the different data medias and that broadcast methods are available. ConCom can then be implemented on these network protocols as a semantic communication layer to support ad hoc features and the design of Ubicomp applications.

3.4 Security Issues in ConCom

The first impression of the ConCom approach is that it breeds a bunch of illusive security issues. Here are some possible questions that came to mind, while considering the ConCom pen application:

Confidentiality & Privacy: what happens when either the subject or the object of the payload has sensitivity associated? – "I am a pen, I am writing your credit card number…."

Integrity, Authenticity and Non-repudiation: how can a receiver prove the authenticity of the sender? – “Is this really my pen?” “This pen wrote my credit card number and I want to prove it!”

Authorization and Access Control: how does a sender formulate an access request and how does a receiver specify authorizations? Alternatively, can the sender specify which receivers can subscribe to its data?

In spite of the validity of these concerns, ConCom does not really pose new challenges for security. The issues for security remain with the technical capability of the underlying devices (i.e. can the devices process a standard symmetric or asymmetric crypto algorithm?) as well as with the sensitivity attributed to the data and applications it serves. ConCom does not invalidate the likelihood that a sender and receiver have a pre-existing relationship, which allowed them to exchange a secret or accept a public key. This may be part of the initialization process of an application that uses ConCom as its operational communications protocol. Secondly, the sensitivity of a message can lie within the subject, the attributes, the action or the entire message. Therefore, adapting security for ConCom should also allow a choice to either encrypt parts of the message or the entire message, having weighed the tradeoffs for application performance and communications latency – consider IPSEC’s different modes of AH (Authentication Header) or ESP (Encapsulating Payload) [8]. To summarize this, established crypto algorithms to maintain confidentiality and privacy of data, as well as to assert integrity, authentication and non-repudiation can still be applied. There are however new approaches to addressing the resource (bandwidth, processor time, battery power) consumption and limitation issues [9,10]. In terms of authorization and access control requests, we can consider that a “request for access to a service” is the action element of a ConCom sentence. Therefore, a subject requesting access to a target would create a ConCom sentence with the following components: “I am <Subject>, with attributes <Credentials>, requesting access to <Target>. The policies do not need to be changed, only the introduction of a component that translates a ConCom sentence into a standard service request.

4 Implementation and Applications

AwareCon [11], the ad hoc protocol for the Smart-Its Particle Computers [5] uses ConCom exclusively for its data traffic. Smart-Its Particle Computers are small embedded networked devices with limited power and computing resources. The structure and traffic flow defined in ConCom has been fully implemented and proved to be a suitable way of application design. The parts of the sentences are mapped as follows: One ConCom sentence is represented by one packet on the AwareCon network. It is therefore basically limited to payload size of one packet in AwareCon, typically 64 Bytes, but can be extended through fragmentation techniques. Every packet is a new sentence and therefore no extra implementation of the “full stop” is necessary to delimit a sentence. The subject is represented by 3 alphanumeric characters encoded in 16 bits. For example a doorplate subject is referred to as “ADP” meaning “Artifact Door Plate”. Another example is the virtual network manager, which represents any transmission to manage and adjust network parameters in AwareCon. It is called

“ACM”, meaning “Artifact Control Management”. After the subject, a number of data units follow, which represent the attributes to be transmitted. These data units are all preceded by a type representing the attribute descriptor, which again are encoded with 3 alphanumeric characters to 16 bit. The types represent actions taking place or attributes like sensor values that correspond to the subject. As an example, we show how the ConCom sentence “I am a TEMP2000, the temperature is 35 degree Celsius” looks when transmitted in the AwareCon implementation: <AT2,STE,35>. AT2 as the subject stands for “artifact TEMP2000”, STE as the attribute descriptor for “sensor temperature”, and 35 as the attribute value represents the temperature.

The most important feature that we could gain with using ConCom as the communication basis in AwareCon is what we call the *early shutdown mechanism*. Incoming packets to a Smart-Its Particle are preceded by the subject description, which is transmitted first. A receiving network instance can immediately check the subscriptions when a packet arrives. If the subject wasn’t found in the subscription list, the packet reception is interrupted and the receiver components of the device are switched to sleep mode to save energy. They are turned on again when the transmission of the “uninteresting” packet is terminated and the next packet follows. During the sleep time of the RF transceiver, the total power consumption on Smart-Its Particles is reduced from ca. 13mA to ca. 5mA. AwareCon is a slotted protocol and the additional time that a receiver can shut down its RF transceiver using the early shutdown mechanism is around 40% of the slot time. Those two results together and the assumption, that only 10% of the traffic is of interest reduce the energy consumption by 30%! More flexible hardware architectures that can adjust processors activity (which is not possible on the Particles) could easily achieve 50% and more power reduction. This reasonable reduction of power consumption is caused by the structure of the ConCom language and does not introduce any packet loss.

As a second advantage, the ConCom language supports developers of application for networked embedded devices. If devices are brought into a heterogeneous environment where a lot of applications run at the same time, the filtering of messages is necessary and would normally be done with parsers. With ConCom, every network node can easily define its interest and filter messages in a fast an efficient way.

4.1 Context Nuggets

Context Nuggets [12], an interactive game based on Smart-Its Particle Computers, takes advantage of the ConCom structure and is a real ad hoc application. The players of the game carry a Particle Computer with sensors. When sensor values have been gathered, they are formed into units and have to be exchanged with other players. The wireless link is reduced to a one-meter range. That means that players of the game have to meet each other to exchange sensor value units on a one-to-one basis. The player with the most collected sensor values wins. The exchange of values is done automatically by the Particle Computers when communication partners come in range.

Therefore, the game software running on the mobile nodes quickly has to find communication partner that are in range. All nodes that carry the software for the game are considered as subjects "Alchemist" (the name comes from the story of the game). All a network node has to do is to subscribe locally to "Alchemist" and then all packets on the wireless channel are filtered. Through this, the development of the game software was simplified. Other applications running at the same time do not complicate the reception of the interesting packets or produce more effort in the mobile nodes like parsers would. The partner discovery is not done through a service discovery process, but through an application based content filtering that is fast enough for mobile and ad hoc settings, introduces very little overhead and is easy to understand and to program.

4.2 AwareOffice

The AwareOffice [13] project at TecO focuses on augmenting office environments with ubiquitous computing technology. We develop a collection of applications making use of ConCom as communication language. Augmentation of the environment and the artifacts in the AwareOffice is realized by Particle Computer devices embedded into chairs, tables, windows, doors, whiteboard pens, etc. Additionally the environment contains networked devices including an interactive doorplate and digital cameras. By using ConCom for communication of context information in the AwareOffice we reduced the effort required to set up a service discovery mechanism for the environment. Artifacts that provide information about their context or that of the environment, simply broadcast this information back to the environment. Applications interested in certain contexts add these to their subscription lists and receive the intended data from that point on. ConCom also supports the derivation of context in multiple steps and communication of the resulting contexts of differing abstraction levels. This is achieved by exploiting the flexibility of the subjects and attributes in a ConCom sentence. E.g. a whiteboard pen can communicate its context with the sentence "`I am Stylograph 2000, someone is writing`" as described in section 3. In the AwareOffice a digital camera is subscribed to the pen as a subject and takes a picture of the whiteboard whenever the pen's context changes from "someone is writing" to "the writing stopped" and places the picture in a folder associated with the ongoing meeting. In turn the camera can provide a context that is derived form its state but has not necessarily the camera itself as subject. A typical example for this is a context like "a meeting is going on at the moment" that the camera could provide on basis of the "writing / not writing" contexts it received from the pen such that other applications that are subscribed to the subject "meeting room" can receive that information. We do not however go into the detail of the complex context derivation process. In this case the meeting room is a virtual subject, for all artifacts that provide data on the context of the meeting room can use that as the subject of a ConCom sentence they send. This easily implements a multistage process of knowledge aggregation in ubiquitous computing environments.

5 Related Work

There exist research in wireless sensor network communication protocols, which show strong relationship to our work. Two initiatives we selected are the Directed Diffusion [14] and SPIN [15] protocols, where the authors describe protocols for information dissemination with particular emphasis on energy resources of participating devices. Both protocols are data-centric with regards to the naming schema, where, for example, sensors are coupled with the data values.

In the Directed Diffusion protocol this naming consists of attributes-values-pairs describing a task. Named data is propagated though the wireless sensor network in order to advertise an interest in such data and invoking a remote task for generation of this data. However, in contrast to ConCom, the transport of subscribed data is done in a destination-address manner, as the subscription on the remote source node only selects the data that should be sent to the device address of the requesting network node, which propagated the interest.

SPIN proposes an "efficient dissemination of individual sensor observations to all the sensors in a network, treating all sensors as potential sink nodes"[15, pg.1]. In SPIN, data is named by a preceding high level data descriptor. Dissemination works through a three-way handshake. New data is advertised with only the descriptor being sent and interested nodes can then query the new advertised data. The SPIN protocols are designed for loss less networks with focus on energy efficient dissemination. They do not use broadcast on MAC-layer. ConCom proposes a powerful representation through the usage of sentences and generally uses broadcast and RAUM for the dissemination. ConCom does not focus on energy efficiency but fast and simple distribution of information in ad hoc settings.

Already in the early 90s the authors of [20] suggested protocols that include information and requirements of the presentation layer to optimize lower network layers and traffic. They propose the principle of application layer framing (ALF) where network protocols use packets which serve and are meaningful to applications. ConCom is influenced by this ideas and introduces semantic data traffic for ad hoc and heterogeneous settings.

TinyOS, an operating system for the Berkeley Motes [2], implements the concept of Tiny Active Messages [16]. In this concept a message invokes on a sensor node a message handler selected by an identifier at the beginning of the message and the remaining data of the payload are passed as parameters for this handler. Although Tiny Active Messages share a similarity with ConCom, our approach goes further by allowing the creation of sentences. This allows the representation in which context the data or other context originated and therefore allows a very flexible handling of a ConCom message.

Location beacons like the IR beacons from CoolTown [17] or the Xerox Parc beacons [18] are used in order to support mobile devices in the finding of their locations. Thereby the beacons broadcast a message which identifies the location of the beacon. Receiving devices must then further resolve this identification in order to obtain the location information. Information broadcasted in ConCom is much more expressive and does not necessarily require further resolving. In particular, location information in RAUM can be supplemented by additional information like accuracy, so that ConCom enabled applications can make use of it if necessary.

6 Summary and Outlook

ConCom provides an easy to understand way to efficiently distribute context and data in Ubicomp setting. It is resource efficient and works connectionless and without addressing. ConCom is affiliated with the RAUM system to handle semantic location addressing and routing to support applications beyond ad hoc settings in one RF cell. The light weighted implementation in AwareCon produces reasonable power savings and the ContextNuggets game and AwareOffice show the advantages of using ConCom for the development and use of ubiquitous computing applications.

As a next step we will investigate the building of sentences using a hierarchy of subjects. In this understanding, subscriptions to subjects could be inherited to their subordinated subjects to realize more flexible subscriptions on a semantic level. A subscription to parent subject would include the subscription to all child subjects. As the next consequence, an optimal ConCom system would filter incoming sentences according to their transported *context* itself. For this final goal, more work is necessary to understand what context means to applications and how it can be clearly represented.

References

1. Gellersen, H-W., Beigl M., Krull, H.: The MediaCup: Awareness Technology embedded in an Everyday Object, 1th Int. Sym. Handheld and Ubiquitous Computing (HUC99) 1999, Karlsruhe, Germany
2. Hill, J., Culler, D.: Mica: A wireless platform for deeply embedded networks, IEEE MICRO Journal, November-December 2002.
3. Website of the EYES project: http://eyes.eu.org/ [Accessed: 03/03/04]
4. Rhee, S., Liu, S.:An Ultra-low Power, Self-organizing Wireless Network and Its Applications to Non-invasive Biomedical Instrumentation, IEEE/Sarnoff Symposium on Advances in Wired and Wireless Communications 2002, West Trenton, USA.
5. Beigl, M., Zimmer, T., Krohn A., Decker, C., Robinson, P.: Smart-Its - Communication and Sensing Technology for UbiComp Environments. Technical Report ISSN 1432-7864 (2003)
6. Jini (TM) Technology Core Platform Specification, Available on http://www.jini.org/standards, [Accessed 03/03/04]
7. Beigl, M., Zimmer, T., Decker, C.: A Location Model for Communicating and Processing of Context, Personal and Ubiquitous Computing Vol. 6 2002 Issue 5-6, pp. 341-357
8. Maughan, D., Schertler M., Schneider M., Turner J.: Internet security association and key management protocol (isakmp)". Internet-draft, IPSEC Working Group, June 1996.
9. Perrig A., Canetti R., Tygar D., Song D.: The TESLA Broadcast Authentication Protocol. RSA CryptoBytes, 5(2):2--13, 2002

10. Karri R., Mishra P.: Optimizing IPSec for energy-efficient secure wireless systems, System-level Power Optimization for Wireless Multimedia Communication, Kluwer Academic Publishers, April 2002
11. Beigl, M., Krohn, A., Zimmer, T., Decker, C., Robinson, P.: AwareCon: Situation Aware Context Communication, Ubicomp 2003, Seattle, USA
12. Beigl, M., Krohn, A., Decker, C., Robinson P., Zimmer, T., Gellersen, H., Schmidt, A.: Context Nuggets: A Smart-Its Game, Adjunct Proceeding of Ubicomp 2003, Seattle USA; http://www.teco.edu/contextnuggets [Accessed 03/03/04]
13. The AwareOffice Initiative at TecO [Online] TecO, University of Karlsruhe, Germany, Available from: http://www.teco.edu/awareoffice [Accessed 03/03/04]
14. Intanagonwiwat, C., Govindan, R., Estrin D.: Directed diffusion: A scalable and robust communication paradigm for sensor networks. In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCOM '00), Boston, Massachusetts, August 2000.
15. Kulik, J., Rabiner W., Balakrishnan, H.: Adaptive Protocols for Information Dissemination in Wireless Sensor Networks, Proc. 5th ACM/IEEE Mobicom Conference, Seattle, WA, August 1999
16. Hill,J.: A Software Architecture Supporting Networked Sensors. Masters thesis at the University of Berkeley, December 2000
17. Kindberg, T., Barton, J., Morgan J., Becker G., Bedner, I., Caswell, D., Debaty, P., Gopal, G., Frid, M., Krishnan, V., Morris, H., Pering, C., Schettino, J., Serra, B., Spasojevic, M.: "People, Places, Things: Web Presence for the Real World". In the Proc. of the 3rd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2000), Monterey, USA, Dec. 7-8 2000.
18. Want, R., Fishkin, K.P., Gujar, A., Harrison, A.L.: Bridging Physical and Virtual Worlds with Electronic Tags. Conference on Human Factors in Computing Systems (CHI99), Pittsburgh, Pennsylvania, USA, May 15-20, 1999
19. Goland, Y., Cai, Ting., Leach, P., Gu, Y., Albright, S.: Simple Service Discovery Protocol 1.0, October 1999. Available on http://www.upnp.org/resources, [Accessed 03/03/2004]
20. Clark, D., Tennenhouse, D.: Architectural Consideration for a New Generation of Protocols. ACM SIGCOMM 1990