



A context management system for a cost-efficient smart home platform

J. Schneider, A. Klein, C. Mannweiler, and H. D. Schotten

University of Kaiserslautern, Paul-Ehrlich-Street, Building 11, Germany

Institute for Wireless Communications and Navigation

Correspondence to: J. Schneider (schneider@eit.uni-kl.de)

Abstract. This paper presents an overview of state-of-the-art architectures for integrating wireless sensor and actuators networks into the Future Internet. Furthermore, we will address advantages and disadvantages of the different architectures. With respect to these criteria, we develop a new architecture overcoming these weaknesses. Our system, called Smart Home Context Management System, will be used for intelligent home utilities, appliances, and electronics and includes physical, logical as well as network context sources within one concept. It considers important aspects and requirements of modern context management systems for smart X applications: plug and play as well as plug and trust capabilities, scalability, extensibility, security, and adaptability. As such, it is able to control roller blinds, heating systems as well as learn, for example, the user's taste w.r.t. to home entertainment (music, videos, etc.). Moreover, Smart Grid applications and Ambient Assisted Living (AAL) functions are applicable. With respect to AAL, we included an Emergency Handling function. It assures that emergency calls (police, ambulance or fire department) are processed appropriately. Our concept is based on a centralized Context Broker architecture, enhanced by a distributed Context Broker system. The goal of this concept is to develop a simple, low-priced, multi-functional, and save architecture affordable for everybody. Individual components of the architecture are well tested. Implementation and testing of the architecture as a whole is in progress.

1 Introduction

Today, the importance of context information has increased and a lot of systems are already based on context data. Especially automated system, like Home Automation or Home Assisted Living are based on such data. Therefore it is necessary to develop appropriate context management architectures as well as concepts and mathematical methods to inter-

pret the data and adapting the system. In this paper, we introduce a cost efficient Context Management Platform which is based on distributed Context Broker (CxB) architecture for enabling context aware system adaptation. But first it is important to define the notion "Context". According to Dey and Abowd (Dey, 1999), Context is "any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and application themselves."

For the so called Smart Home Platform (SHP), context consists of sensor data and data which is gathered from the internet (e.g. weather). Consequently our architecture must be able to handle all these kind of context data. Different kinds of architectures have been studied in the last years. Most of the architectures have been developed with a centralized broker architecture, e.g. CoBrA (Chen, 2003a) (Chen, 2003a) or (Schneider, 2010a).

Easy Meeting (Chen, 2004a,b) and MyCampusEasy (Gandon, 2003) (Gandon, 2004) are typical projects were researchers deal with reasoning. Easy Meeting is based on the Context Broker Architecture (CoBrA) and is a prototype of an intelligent meeting room. The Carnegie Mellon University has developed the MyCampus to provide context aware services for the community in the university. The e-Wallet is the key element, that supports access control and obfuscation rules for user's privacy.

The introduced approaches are based on a centralized Context Broker. A resulting disadvantage of such a system is the limited quantity of administrable clients. However, the basis for our system will be a centralized Broker system (compare CCast project, CCast, 2009). It breaks the system down into three main entities, a Context Provider (CxP), a Context Consumer (CxC) and a Context Broker. The CxB acts a broker for all kind of data and is aware of all CxPs and CxCs. Figure 1 shows the principal structure of the centralized CxB approach. This approach is not entirely new and

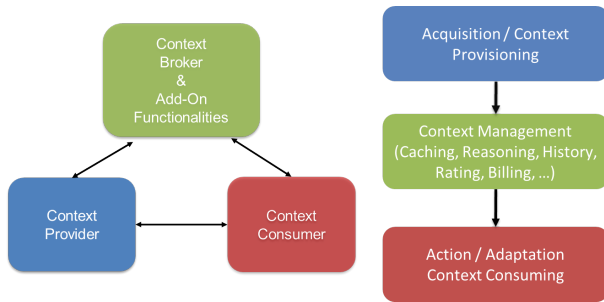


Fig. 1. Centralized Context Management architecture.

has already been studied by several researchers and used in different projects.

In the case of large architectures, more precisely systems with hundreds of clients, an efficient management of the data is needed. A disadvantage of the centralized approach is the decreasing performance if the number of entities increases. Especially in the use case of a Smart Home Platform, sensor data and other context data (e.g. from the internet) will be used to trigger events. Additionally, the system intelligence is located into so called Services. Each Service implements a Context Consumer and a Context Provider Interface, furthermore it has an interface to subscribe for context data. This interface is needed to enable value-triggered events.

Hence a centralized architecture will not be able to handle all this amount of data. To overcome this weakness, we developed a distributed architecture which is based on the federation of sub context managements systems. The organization of these systems will be comparable to a Domain Naming Service (DNS) and was introduced in (Schneider, 2010b).

2 Decentralized context management architecture

In this section, we introduce our decentralised system approach in detail. But first it is necessary to define some requirements for the system:

- *Plug and Play*: will be fulfilled with respect to Schneider, 2010b and Schneider, 2009
- *Security and Privacy*

2.1 Interfaces

As basic concept we used the centralized broker architecture from (CCast, 2009). Figure 2 shows all necessary interfaces to deploy the system in a centralized manner.

As already explained, for the smart home platform use case a decentralized architecture is necessary. Therefore,

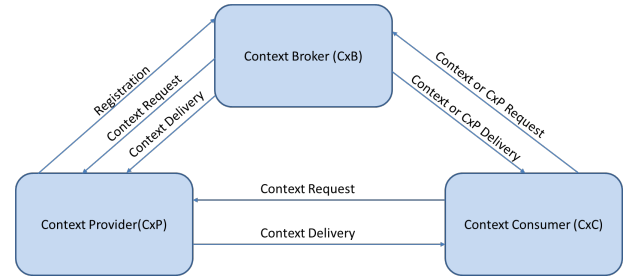


Fig. 2. Interfaces Centralized Context Management architecture.

Table 1. Interfaces of Centralized Context Broker.

Interfaces	Description	Data
<i>Registration</i>	Registration of available CxPs and verifying of the CxP list.	Bidirectional via TCP/IP
<i>Context Transmission with CxP</i>	Request and transmission of Context Data between CxP and CxB	Bidirectional via TCP/IP
<i>Context Transmission with CxC</i>	Request and transmission of Context Data between CxC and CxB	Bidirectional via TCP/IP

some of the available entities need to be enhanced with additional interfaces and will be described in the following subsections.

2.1.1 Context broker CxB

The centralized CxP implementation comprises of the following interfaces:

To use Context Broker in a distributed architecture, additional interfaces are necessary. One interface is needed to enable communication between two CxBs. But before this communication can be established, a CxB detection needs to be performed. Therefore, a Broker detection interface has been included. It is used to detect adjacent CxBs and to request their address to establish a communication. Additionally, each CxB has to provide an Provider Detection interface and a Consumer Detection Interface.

2.1.2 Context provider CxP

A Context Provider delivers various kind of data. It is not restricted to sensor data from physical sensors (such as temperature or light), but also allows to provide pre-processed data, for example system load or available bandwidth. For the centralized system, a CxP is equipped with the Interfaces listed in Table 2.

Table 2. Interfaces of Centralized Context Provider.

Interfaces	Description	Data
<i>Registration</i>	Registration to a CxB.	Bidirectional via TCP/IP
<i>Context Transmission to CxB</i>	Request and transmission of Context Data between CxP and CxB	Bidirectional via TCP/IP
<i>Context Transmission with CxC</i>	Request and transmission of Context Data between CxP and CxC	Bidirectional via TCP/IP

Table 3. Interfaces of Centralized Context Consumer.

Interfaces	Description	Data
<i>Context Transmission to CxB</i>	Request and transmission of Context Data between CxP and CxB	Bidirectional via TCP/IP
<i>Context Transmission with CxC</i>	Request and transmission of Context Data between CxP and CxC	Bidirectional via TCP/IP

With respect to the plug and play requirement, an additional Broker Detection Interface is necessary. This Interface is the counterpart to the Provider Detection Interface in a CxB. The detailed function of a CxB detection is described in Schneider, 2010a.

Additionally each Broker need an interface to communicate with other CxBs. But before a CxB is able to communicate with another CxB, a detection process is to be carried out. Therefore, a CxB sends a broadcast with its own address and waits for response.

Another issue is the implementation of a context subscription interface for enabling services. This interface must be able to allow CxCs to subscribe to context for periodic as well as event-triggered data transfer.

2.1.3 Context consumer CxC

A Context Consumer is an entity which uses context data for its actual purpose. This can be, for example, an actuator for steering a roller blind or the heater. But for developing of our system architecture, we didn't make any restrictions. This offers the possibility to include any kind of services into the platform. For the centralized approach, a CxC is equipped with the Interfaces, listed in Table 3.

This entity also needs an additional Broker detection interface, which is counterpart of the Consumer Detection Interface, located on a CxB.

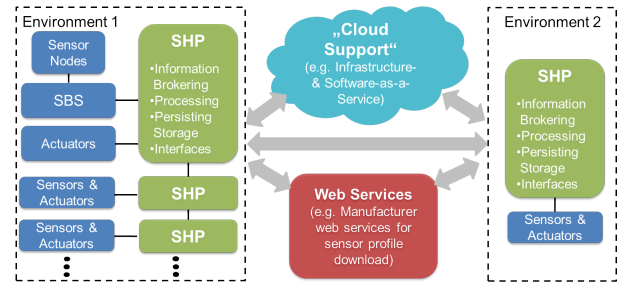


Fig. 3. Cloud-/Web-based Service Provisioning.

2.2 Services

An important issue is the integration and provision of services. Services are software components to implement any kind of functionality. Figure 3 shows an example the principal structure of providing cloud-/web-based services to distributed environment.

With this system, a user is able to add or remove functionalities to the system without any additional hardware. Further more functionalities can be moved to the “Cloud”, in other words, external servers provide most of the necessary services and the user needs only the basic system. Further functionalities can be bought or rented if required.

A user also has the possibility to provide own sensor data or offer own services for public use. With this concept the privacy of a user is ensured and he can decide, which data/information should be private or public.

As already mentioned, the centralized broker architecture is not new and has been studied in several projects. Services offer the possibility to maintain various kinds of tools. A user can subscribe to a service and adjust the service behaviour according to his demand. Therefore the service has to request or subscribe for the necessary context information. With this data, a service is able to trigger an event based on the users adjustments. Each event will request the address of the appropriate CxC and query for the requested data. All kind of data transmission is performed via TCP/IP in a XML coded scheme.

This approach provides the possibility to move software components, especially computationally intensive processes to the “Cloud”. For our system, it doesn't matter how the cloud looks like, or is implemented, respectively. We handle it as a black box but in most cases it will be implemented as a software-service on an external server located in the internet i.e. in the “Cloud”.

2.3 Data transmission

An example for data transmission is given in Fig. 4. It shows the automatic detection of a Smart Home Platform (SHP).

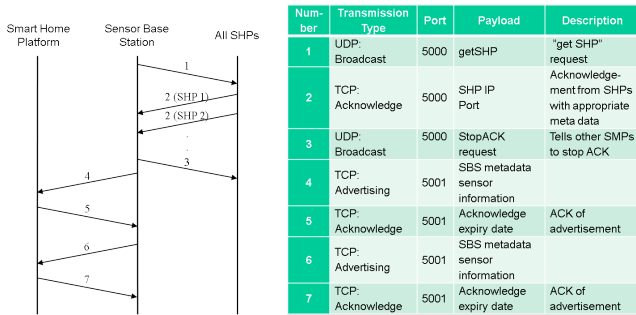


Fig. 4. Automatic SHP detection.

As already explained, all the communication between the entities will be done by a XML derivate Language called ContextML (CML). In this paper we will not provide details on CML, however, refer interested to (Knappmeyer, 2010).

2.4 Context prediction and situation detection

To make a Smart Home Platform as attractive as possible for a consumer, most of the functions should be automatically performed and installation effort should be as low as possible. To deploy such a SHP in a fully automated manner, it is necessary to detect situation or to predict them. Consequently it is necessary to predict context information and to combine the data for gathering new findings. To achieve this goal, different concepts are available to recognize a situation. But first, we define the term "situation": "A situation is an abstraction for a pattern of observations made by a distributed system such as a sensor network" (Cardell, 2010). Therefore different recognition algorithms are available:

- Finite State Machine
- Tree-based evaluation
- Distributed Commitment Machines

The goal of such a complex system is the Combined observations of the individual sensor nodes yield spatial-temporal map of the monitored environment. With this data a service is able to adopt the system behaviour to the users preferences.

To detect a dynamic situation, it is essential to select an appropriate modelling method. Two approaches are adequate for this usage:

- Construction of appropriate Bayesian networks
- Consideration of approximate inference methods (e.g. Monte Carlo algorithm)

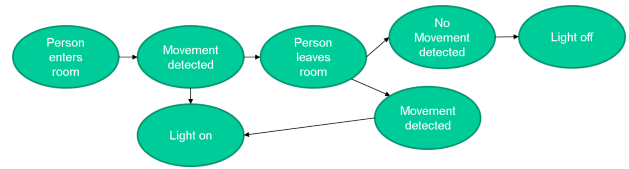


Fig. 5. Bayesian Network.

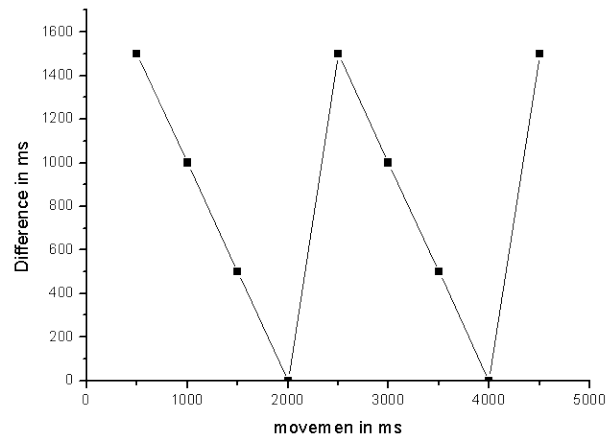


Fig. 6. Room light simulation.

Figure 5 shows an example for a Bayesian network and models the light automation. The room is equipped with different kind of sensors (e.g. movement, temperature, ...), consequently the system is able to detect if somebody is in the room. If movement detection is true, the light is switched on. A weakness respectively a problem is the time how long the room is in use. This results in the problem, how long should we turn on the light after a movement? If there is no movement, has the person left the room or is she/he still sitting inside the room? If we describe the system with a fixed on-time, we get the graph in Fig. 6.

To get an energy efficient system, the time difference between the time of movement and the light-on time should be minimized. Figure 6 shows that systems with fixed times are not able to provide satisfying results. Furthermore general concepts to provide adaptive system behaviour need to be developed. In case of a smart home platform, most of the system parameters depend on the users behaviour and are function of time. To ensure such a behaviour, we use a Finite State Machine and combine this with a Bayesian Network. The following subsections will describe our model in detail.

2.4.1 Finite state model

A Finite State Model is a mathematical model that is used to design computer programs and digital logic circuits. As already explained above, the system behaviour is simplified into a finite amount of states. A transition from one state to another can only be initiated by a triggering event or a changing condition. As a result, we sustain a model, similar to Fig. 5.

2.4.2 Bayesian network

In our case, Bayesian Networks can be seen as an extension of the finite state model. The additional part or the extensions are the conditional probability tables. To make such a system adaptive respectively intelligent, these tables are not fixed and must depend on external parameters. In our movement example, time seems to be a very important parameter because most of the human behaviour depends on the daytime. Therefore appropriate models for the user's behaviour have to be found.

3 Conclusions

In this paper we introduced the so-called Smart Home Platform and listed the requirements to the Context Management Platform. Furthermore, we have shown that it is possible to extend available centralized broker architectures to distributed architectures with adding interfaces to the available components. A benefit of this approach is that available interfaces can be used without any change. The new introduced interfaces can use the available communication protocol which is based on ContextML. Additionally the architecture has been extended with services that offer the user to add functionalities to the system without any change of the hardware.

Furthermore, we have shown approaches to predict context and to detect situations. This enables an intelligent behaviour of the Smart Home Platform.

Acknowledgements. This work has been supported by the Federal Ministry of Education and Research of the Federal Republic of Germany (Foerderkennzeichen 01 BK 0808, GLab). The authors alone are responsible for the content of the paper.

References

- Dey, A. K. and Abowd, G. D.: Towards a Better Understanding of Context and Context-Awareness, College of Computing, Georgia Institute of Technology, Tech. Rep. GIT-GVU-99-22, 1999.
- Context Casting C-CAST, available at: <http://www.ICT-CCAST.eu>, Last access: October 2011.
- Chen, H., Finin, T., Anupam, J.: Semantic Web in a Pervasive Context-Aware Architecture, *Artificial Intelligence in Mobile System*, 2003a.
- Chen, H. and Finin, T.: An ontology for context-aware pervasive computing environments, *The Knowledge Engineering Review* page 197–207, 2003b.
- Chen, H., Finin, T., Joshi, A., Kagal, L., Perich, F., and Chakraborty, D.: Intelligent agents meet the semantic web in smart spaces, *IEEE Internet Computing* page 69–79, November–December 2004a.
- Chen, H., Perich, F., Chakraborty, D., Finin, T., and Joshi, A.: Intelligent agents meet semantic web in a smart meeting room, *AA-MAS 04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 854–861, IEEE Computer Society, Los Alamitos CA USA, 2004b.
- Gandon, F. and Sadeh, N. M.: A semantic e-wallet to reconcile privacy and context awareness, *Lecture Notes in Computer Science: The Semantic Web – ISWC 2003*, 385–401, 20–23 October 2003.
- Gandon, F. and Sadeh, N. M.: Semantic web technologies to reconcile privacy and context awareness, *J. Web Semant.*, 241–260, 2004.
- Schneider, J., Klein, A., Mannweiler, C., and Schotten, H. D.: Environmental Context Detection for Context-Aware Systems, *ICST's Global Community Magazine*, 2010a.
- Schneider, J., Mannweiler, C., Klein, A., and Schotten, H. D.: Einbindung von Sensoren und Sensornetzwerken in das Future Internet, in *ITG-Fachbericht der 15. ITG Fachtagung Mobilkommunikation*, Osnabrück, Germany, May 2010b.
- Schneider, J., Mannweiler, C., Klein, A., and Schotten, H. D.: Erfassung von Umgebungskontext und Kontextmanagement, *ITG-Fachbericht Band 215 der 14. ITG Fachtagung Mobilkommunikation*, Osnabrueck, Germany, May 2009.
- Knappmeyer, M., Liaquat, S., Kiani, F., Moltchanov, B., and Baker, N.: ContextML: A Light-Weight Context Representation and Context Management Schema, *IEEE International Symposium on Wireless Pervasive Computing (ISWPC)*, Modena, Italy, May 2010.
- Cardell-Oliver, R. and Liu, W.: Representation and Recognition of Situations in Sensor Networks, *Communications Magazine*, IEEE The University of Western Australia, March 2010.