Electronic Communications of the EASST
Volume 27 (2010)

Workshop über
Selbstorganisierende, adaptive, kontextsensitive verteilte Systeme
(SAKS 2010)

Building an Infrastructure Level Context Model in Ambient Assisted Living

Manfred Wojciechowski

13 Pages

Guest Editors: Klaus David, Michael Zapf
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Building an Infrastructure Level Context Model in Ambient Assisted Living

Manfred Wojciechowski

Fraunhofer Institute for Software and Systems Engineering

Abstract: Ambient Assisted Living (AAL) services provide intelligent and context aware assistance for elderly people in their home environment. This domain puts special requirements on context modeling that are not in the scope of current context modeling approaches. Such an approach has to support all phases of an AAL service, from its specification and development until its operation within the user's smart home environment. In these phases different types of context models can be identified. We have developed a comprehensive context modeling approach for the development of AAL services. Part of it is the separation of context modeling into infrastructure, service adaptation and end user modeling specific aspects. In this paper we focus on the infrastructure, which includes the context sensors available in the smart home environment. Therein we present our context modeling approach starting from a conceptual context model. We also introduce a context management system based on a metamodel that supports its seamless transition into an operative context model without further implementation.

Keywords: Ambient Assisted Living, Smart Home, Context Modeling, Infrastructure, Conceptual Context Model, Operative Context Model, Context Management System

1 Introduction

‚Ambient Assisted Living' (AAL) aims at extending the time that the elderly can live in their home environment independently. A smart home environment integrates into the living space of the inhabitant and provides services that help to increase their autonomy and gives assistance in different activities of daily life. Key technologies for AAL services can be found in the research areas of 'home automation' and 'ambient intelligence.' Examples of such services can be found in [1].

Home automation is focused on the development of sensors, actuators and smart appliances that can be integrated into a home network. A home automation infrastructure, e.g. OSGi [2], can then be used to interconnect these devices and to provide services.

'Ambient Intelligence' follows the goals of a vision expressed by Marc Weiser [3]. In that vision the computer becomes invisible for the user, enriches his natural environment with additional intelligence and supports him in his daily goals. Communication with the intelligent environment happens intuitively by interface support for language, movement, gesture and pointing [4]. Additionally, context awareness can be used to observe the inhabitant and his environment and to provide services that adapt accordingly without the need for explicit user interaction.

In the project 'SmarterWohnen' [5], we have implemented and tested a number of AAL services together with a local housing company. They have been deployed in apartments, which have been equipped with different sensors and actuators, and are now used by a number of selected tenants. These services include intrusion detection, water and gas leakage detection, health related services and other various home automation services. We have developed an AAL service platform which also includes a context subsystem. This subsystem supports the integration of context sensors, the refinement of context information and the provision of a layered context model.

We have developed a context modeling approach that supports the special requirements of the development of context aware AAL services. The approach includes the definition of different context model types along the life phases of an AAL service and the transitions between them. Another important result is the separation of context modeling into three separate layers. An important aspect that is covered by one of the layers is the description of the smart home environment including the available context sensors. Our context modeling approach is based on the definition of a metamodel covering the three layers. We have developed a context management system, which fully implements the context metamodel. A concrete context model is provided by an XML model description without further implementation, which speeds up the development process for context aware applications. The paper will give an overview on our context modeling approach. Therein we'll focus on the infrastructure layer and demonstrate the development of its conceptual context model and the transition into operation using the context management system.

The rest of the paper is organized as follows. In section 2 we describe the state of the art in context modeling and existing frameworks. In section 3 we then motivate our approach from the requirements that can be identified from the development of context aware services in AAL. Then we introduce our three layered AAL context model and describe the infrastructure layer in detail in section 4. This includes the definition of its metamodel and a graphical notation. Section 5 introduces our context management system and describes its usage for the implementation of the context infrastructure. We then end with a conclusion and outlook in section 6.

2 State of the Art

We follow Dey's [6] definition in which 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." Context aware applications can adapt their behavior directly to a situation on the user's environment without having to ask for explicit user input. Context information can be used to adapt the behavior of an application in many ways. A classification on context adaptivity is given in [7]. Examples are location based services, where the information and service supply is dependent on the current location of the user. Additional information on opening hours of restaurants, the time of day, the current activity of the end user or specific situations are further examples for context information that can be used to realize context aware applications.

A context model formally describes the relevant aspects of the real world that are used for an application. It abstracts from the technical details of context sensing and allows coupling the real world to the technical view of context adaptive applications [8]. Therefore context models play an important role for building applications that can react on real world events. Research in context modeling is not new, and there are already a number of approaches introduced in the context awareness community. An overview of actual approaches has been given in [9].

Simple context models can rely on name-value-pairs, e.g. for the annotation of services with context information [10]. Another solution is the use of XML for the definition of context models, which can be used for providing context profiles [11]. Currently, different context models based on ontologies are discussed, e.g. in [12][13]. Ontologies can be used for knowledge sharing in an open and dynamic distributed system and to provide a means to reason about contextual information [14]. Despite the existing approaches, there is not yet a common standard agreed upon for context modeling, though discussion of such a standard is ongoing. Currently, the discussion of context modeling is focused on the implementation of context aware applications. Most approaches are focused on the abstraction of sensor information and the provision of high level context information. An exception is the conceptual context modeling approach described in [15]. This approach defines a graphical notation which can be used to design and discuss a context model independently from its later implementation. A mapping of the conceptual modeling constructs to ontologies is discussed in [16].

Context modeling in AAL is a domain specific view on general modeling approaches. Therefore existing results e.g. from the domain of ambient intelligence [17] can also be applied here. Open questions in context modeling and architectural patterns in general remain also open in this domain, e.g. uncertainty of context information and privacy aspects. Research in AAL is currently focused on the identification of concrete context aware AAL services. In [1] examples of AAL services can be found that already have been realized or are still in discussion. In [18] a classification of context aware services in AAL is given which distinguishes between indoor and outdoor assistance for emergency treatment, autonomy enhancement and comfort services. An agreed on common context model in this domain is not available.

In [14] three different architectural approaches for the implementation of the context related functionality are identified: direct sensor access, middleware infrastructure and context server. In [19] an overview on existing context frameworks and servers and a discussion of their features are given. In [20] the existing approaches are classified regarding their usage of a context model. The authors distinguish between applications with no application-level context model, with an implicit context model or an explicit context model. An explicit context model can be provided by a context server that encapsulates functionalities focused on the inquiry, preprocessing and management of context information. As an example the authors present the ACoMS context management system. Current approaches require and support an explicit implementation of the application specific context models.

Our approach extends the current context modeling approaches to the whole life phases of a context aware AAL service and is not restricted to a certain aspect, e.g. the operative view. We also introduce a context management system that is based on an explicit context metamodel that supports developer definable application specific context models without the need for further implementation.

3 Requirements from AAL

In the following we identify those requirements that result from the vision of an open service marketplace in AAL. First we focus on the life phases of context aware AAL services and the usage of context models therein. Then we identify different context modeling aspects that are of special importance in the AAL domain.

3.1 Life phases of AAL Services

Context aware services in AAL are provided to support the inhabitant of a smart home environment in his current life situation. The set of AAL services that can be supported by the smart home environment depends on the environment's capabilities regarding its actuators and context sensing capabilities. As his life situation changes the inhabitant is in need for a different set of supporting AAL services. The extended smart home capabilities can lead to new AAL services that can be supported. The provision of a service marketplace is a solution for that dynamic nature of AAL services. Such a marketplace serves as a mediator between the service offer of AAL service developers and the demands of the inhabitant of a smart home. Service developers implement context aware AAL services and use the marketplace to publish them. Inhabitants use the marketplace to search for suitable services and integrate them into their smart home environment. The AAL services will then subsequently be used by in the inhabitant. Following this view on the development and usage of AAL services, we can identify different use cases for context modeling with a different focus. Based on this phase model, we define the following types of context models:

- Informal context model: In the phase of the development of a context aware AAL service, the functionality of such a service, including the context adaptive features, has to be defined. Starting with an informal description, these features can be determined in a product concept catalogue.
- Conceptual context model: Based on the informal description of the context adaptive features of an AAL service, a conceptual context model can be derived. This kind of context model abstracts from the implementation details and is used for the discussion and documentation of the relevant context information and their interdependencies.
- Descriptive context model: An implementation independent description of the context aware features of an AAL service is needed in the distribution and deployment phase. It is needed for the registration of a service within the service marketplace and to give the end user information on the functionality and the context aware features of such a service.
- Operative context model: An operative context model is needed for the operation and usage of context aware services. It realizes the technical basis for the management and provision of context information. It is implicitly or explicitly implemented by a context infrastructure. It is needed in both the smart home environment of the inhabitant and in the development and testing environment of the service developer. Most context modeling approaches found in literature can be assigned to this model type.

These different types of context models are relevant in different phases an AAL service. There have to be transitions between these model types along the phases. In AAL the different model types and transitions have to be supported.

3.2 Context Modeling Aspects

Independent from the life phase model there are different context modeling aspects that are of special interest in the AAL domain. In AAL the service developer is not the owner of the smart environment and therefore cannot define and build on a standard context infrastructure. A context model therefore is needed to describe the individual smart home infrastructure and the context information that are available therein. This includes the available context sensors and their capabilities. Such a context model has to be extensible, since new kind of context information can be made available through new sensor technologies. The description of the infrastructure also has to include context information about available networks, hardware, software and devices that are needed to support ubiquitous computing features. Different proposals for such information are published, e.g. in [17] [21]. We distinguish between infrastructural and application specific context adaptive behavior of an AAL service. The infrastructural context adaptive behavior is independent from the primary functionality of a concrete service and should be provided by the smart home environment, which can be applied by different AAL services. An example for such a ubiquitous feature is the presentation of a message on a suitable device nearest to the inhabitant.

Application specific context behavior has to build upon a context model that differs from the infrastructure view. Such a context model has to include application specific semantics relevant to the context aware behavior of an AAL service. One example is the distinction of a person according to his role, e.g. whether he is the inhabitant, the neighbor or the doctor. This distinction is not relevant for the description of available context information and the integration of context sensors, but it is important for the definition of the context dependent functionality of a health related AAL service. As described in [18] AAL services can be categorized according to their application domain. We distinguish between the following AAL service types:

- Health related services
- Security related services
- Comfort related services
- Social related services
- Economic related services

Each of these service types has a different need regarding the context model and the semantics therein. Therefore service-type specific context models have to be supported by a context infrastructure.

Another important aspect is that of end user context modeling. Context aware services within a smart home environment have to behave according to the inhabitant's expectations. Developers of such services cannot foresee the behavior that is expected by the end user. If a service continuously acts in an unexpected manner, the inhabitant might feel as if he/she is losing control of his/her home environment, thereby decreasing the acceptance of such services. Therefore it is important to give feedback of the smart home's assumptions about the environment and situation that have led to the services' behavior and to allow the inhabitant to define the context aware behavior of his services. Some concrete requirements in the end user aspect have been identified in [22].

4 AAL Three Layered Context Model

As a result of the different context modeling aspects that we have identified in the section before, we have organized our context modeling approach into three separate layers with a specialized point of view on the infrastructure, the service adaption and user interaction. An overview of the layered context modeling approach is shown in the following figure.

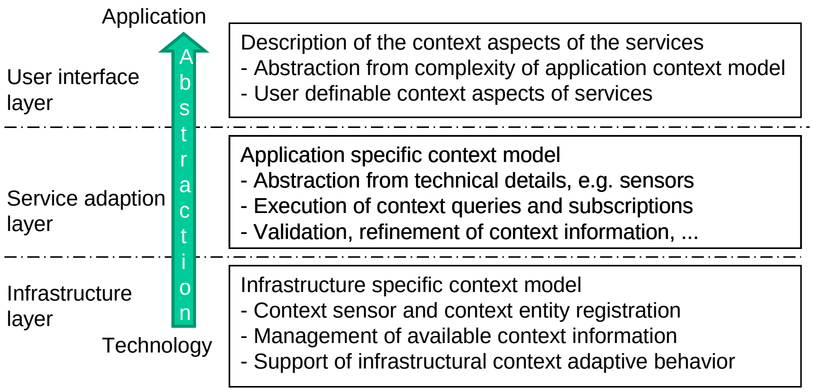

Figure 1: Three layered context model

The context model of the infrastructure layer represents the existing infrastructure resources, e.g. context sensors, and the application independent view on the smart home environment. It is used for the management of the context infrastructure and to support the infrastructural context adaptive behavior. The description of this layer is the focus of this paper.

The context model of the service adaption layer represents the service-type specific context models. This layer abstracts from technical details of the availability of context sensors. It is used to realize the application specific context adaptive behavior of an AAL service. Context information provided by the infrastructure described in the infrastructure layer has to be mapped into the different service-type specific context models. The description of this layer is not part of this paper. Most context models identified and evaluated in [9] can be classified into this layer.

The context model of the user interface layer focuses on the visualization of context aspects to the inhabitant and the definition of the context adaptive behavior of the AAL services by the end user. It abstracts from the complexity of the service-type specific context model. A description of this layer is given in [23]. A description of its application is given in [24].

In the following subsections we describe the metamodel of the infrastructure layer. Then we introduce a graphical notation for the model elements that are relevant for the definition of a conceptual context model of this layer and conclude with an example infrastructure context model.

4.1 Metamodel of the Infrastructure Layer

The metamodel describes the elements that can be used to define a concrete context model. All published context models have an implicit or explicit metamodel which can be specialized on the context modeling domain or based on more general modeling approaches, e.g. an ontology. Some elements in our metamodel also exist in other modeling approaches, e.g. ‘context entity’, ‘context relation’ and ‘context attribute’. We have defined some additional model elements that are needed to define some semantic aspects of context attributes, their representation and implementation, but also the availability of context sensors.

4.2 Graphical Notation

The infrastructure metamodel is the basis for our context modeling approach. Each of the model types has a different focus and therefore a different scope and detail of the metamodel is needed. For the conceptual context model we have defined a graphical notation of the relevant model elements therein. Using this notation a concrete infrastructure layer context model can be defined and documented. The relevant context elements for the conceptual model and their visual representation are described in the following.
A context entity is represented by a two-framed box with the name of the entity type included. An abstract context entity is visualized by a framed ‘A’ included in the box. A specialization relation between two context entities can be defined by an unfilled arrow known from UML.

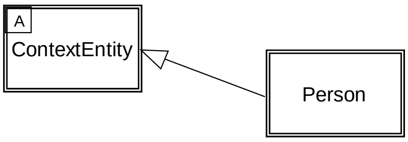

Figure 2: Representation of the model element ‘Context Entity’

A context relation has its own representation visualized by a diamond with the name of the relation included. A context relation is connected to the related context entities using simple arrows. The direction of the arrows indicates the direction of the relation.

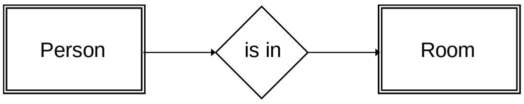

Figure 3: Representation of the model element ‘Context Relation’

A context attribute is represented by a single framed box. It has the names of the dimension and of the attribute included, which are separated by a colon. A number of attributes belonging to the same entity or relation can be gathered in one single box. The representation type of a context dimension can be included in brackets with the name of the dimension. If an attribute has a time reference, then this can also be included in brackets with the name of the attribute. The association of a context attribute to its context entity or relation is visualized using simple arrows.

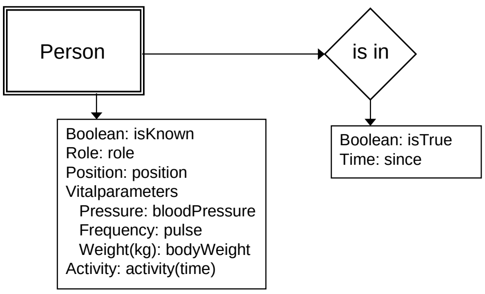

Figure 4: Representation of the model element ‘Context Attribute’

This graphical notation has been inspired by the context modeling approach described in [15]. The modeling dependencies introduced in that approach can be used to define the context rules. From that approach we have removed the classifying association elements, since in our

approach the source and type of the context information is of no relevance for the conceptual context model. These become relevant in the later life phase of the service.
We have discarded approaches like [25] using OWL for defining the conceptual context model, since they do not support the explicit expression of important metamodel elements like the context dimension and representation types or the attributes of context relations and their time reference.

5 Usage of a Metamodel Based Context Management System

We have implemented a context management system that supports our three layered context metamodel. It can be used in the development, deployment, operation and usage phases of the AAL service. It provides an operative context model for each of the three layers. It is part of the context infrastructure of our smart home environment. In the following we present an XML representation of the operative context model that is supported by the context management system. After that, we describe the software engineering methodology for the instantiation of the smart home context infrastructure using the context management system.

5.1 XML-Schema

The context server directly implements our three layered context metamodel. A concrete context model can be described using an implementation-independent representation of the operative context model. We have specified an XML-schema for such a model description. This model description is part of the configuration of the context server. The context server is able to handle this XML description and to transform it to an implementation-specific representation of the context model based on the relational database management system. In the following we give an overview of the XML-schema. For space reason the cardinality information ‘minOccurs’ and ‘maxOccurs’ and the default values on the element definition have been removed from the description.

```
<complexType name="Generalizable"> <sequence>
  <element name="isAbstract" type="boolean"/>
  <element name="parentElementName" type="string"/>
</sequence>  </complexType>
```

Part of the XML-schema is the definition of a complex type ‘GeneralizableElement’. This type describes the properties of a context entity, context relation or context dimension regarding its generalization/specialization features. One or more model elements can be named as the parent of the current model element.
A context dimension is described by the definition of the generalization/specialization features, a textual description, the definition of the default representation and a list of associated representation types. The element “implementation” describes the data type that is used for the implementation-specific representation of the context attribute within the database. We have implemented the following data types: ‘number’, ‘time’, ‘boolean’, ‘coordinate’, ‘enumeration’, ‘taxonomy’ and ‘string’.

```
<element name="ContextDimension">  <complexType>
```

```
  <sequence>
    <element name="generalizable" type="Generalizable"/>
    <element name="description" type="string"/>
    <element name="defaultRepresentation" type="string"/>
    <element name="representation" type="Representation"/>
   </sequence>
   <attribute name="name" type="string"/>
</complexType>  </element>

<complexType name="Representation">
 <sequence>
  <element name="unit" type="string"/>
  <element name="isDescrete" type="boolean"/>
  <element name="isOrdered" type="boolean"/>
  <element name="implementation" type="string"/>
 </sequence>
 <attribute name="name" type="xs:string" />
</complexType>
```

The following example shows the definition of a context dimension ‘Temperature’:

```
<ContextDimension name="Temperature">
  <representation name="celsius">
    <unit>C°</unit>
    <implementation>number</implementation>
  </representation>
</ContextDimension>
```

A context entity is described by the definition of the generalization/specialization features, a textual description and an enumeration of the associated context attributes. The features of the context attribute are also described in analogy to the metamodel and a reference to the context dimension is given.

```
<element name="ContextEntity">  <complexType>
    <sequence>
      <element name="generalizable" type="Generalizable"/>
      <element name="description" type="string"/>
      <element name="attribute" type="ContextAttribute"/>
    </sequence>
    <attribute name="name" type="string"/>
</complexType>  </element>

<complexType name="ContextAttribute">
  <sequence>
    <element name="description" type="string"/>
    <element name="isRange" type="boolean"/>
```

```
    <element name="hasTimeReference" type="boolean"/>
    <element name="isIdentifying" type="boolean"/>
    <element name="isSingleValued" type="boolean"/>
    <element name="isComplex" type="boolean"/>
    <element name="isAscending" type="boolean"/>
    <element name="isDescending" type="boolean"/>
    <element name="contextDimension" type="string"/>
  </sequence>
  <attribute name="name" type="string" />
</complexType>
```

The following example shows the definition of a context entity ‘Person’ with two context attributes:

```
<ContextEntity name="Person">
  <generalizable>
<parentElementName>Contextentity</parentElementName>
  </generalizable>
  <attribute name="isInTreatment">
    <contextDimension>Boolean</contextDimension>
  </attribute>
  <attribute name="bodyWeight">
    <contextDimension>Weight</contextDimension>
  </attribute>
</ContextEntity>
```

A context relation is described by the definition of the generalization/specialization features, a textual description, an enumeration of the associated context attributes and the reference to the related entities.

```
<element name="ContextRelation"> <complexType>
  <sequence>
   <element name="generalizable" type="Generalizable" />
   <element name="description" type="string"/>
   <element name="attribute" type="ContextAttribute"/>
   <element name="relatedEntities" type="RelatedEntities"/>
  </sequence>
  <attribute name="name" type="string"/>
   </complexType>
  </element>
```

```
<complexType name="RelatedEntities"> <sequence>
  <element name="isDirected" type="boolean" />
  <element name="fromEntityName" type="string"/>
  <element name="toEntityName" type="string"/>
</sequence>  </complexType>
```

5.2 Software Engineering Methodology

Using the infrastructure layer of the context metamodel the focus is on describing the existing infrastructure resources and the application independent view on the smart home environment. First a conceptual context model is defined using a visual modeling tool. The focus therein is to identify and organize the context entities and relations and their attributes from a technical perspective of context sensor integration and application independent management of context information. The generalization/specialization relation can be used to organize common set of context attributes into abstract context entities.

In a next step the conceptual context model is then transformed to the implementation-independent representation of the operative context model. This currently has to be done manually using the provided XML-schema. In this step additional details, e.g. the implementation, are added to the context model. The implementation-independent representation of the context model is automatically transformed into an implementation-specific representation by the context server. These steps are visualized in the following figure.

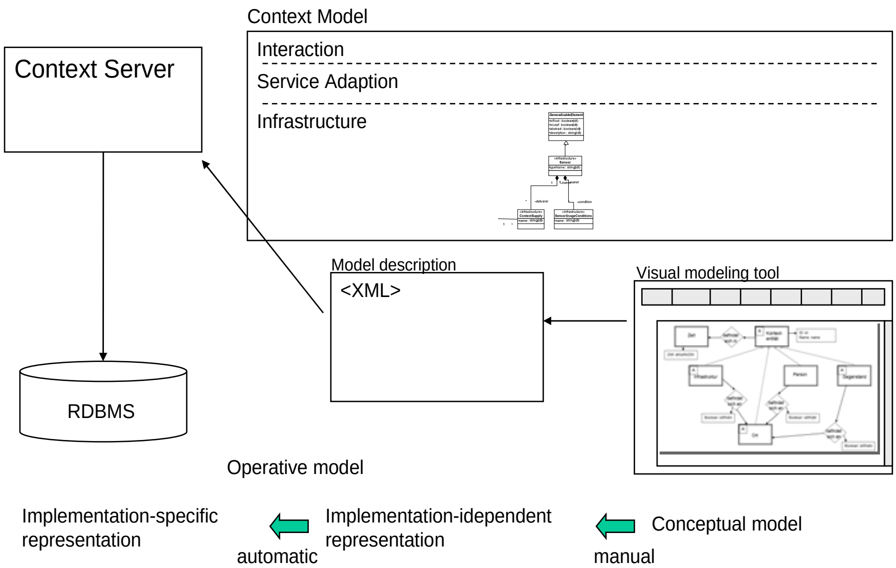

Figure 5: Context Modeling Process

The next step is the preparation of the entity server. The schema for each entity type can be enhanced using a configuration file. A user interface is then provided by the entity server to register a concrete context entity instance.

The last step consists of the preparation of the sensor registry and the registration of the available context sensors. This is also supported using a configuration file and a user interface for the registration process. Having finished these steps the context infrastructure of the smart home environment is prepared. After that the context model of the service adaption and user interaction layer has to be provided, which is not the focus of this paper.

6 Conclusion and Outlook

In this paper we have identified some of the requirements which are specific to context modeling in AAL. Based on our vision of an AAL service marketplace we have defined a life phase model and identified different types of context models that are relevant therein. We have also identified different context modeling aspects that have to be supported by a modeling approach in AAL. These involve the smart home infrastructure, the support of service-type specific context models and user interaction. We have defined a three-layered context metamodel, where each layer is focused on one of these aspects. Each of the context model types has a different focus on context modeling and covers different aspects of the metamodel. Additionally, we have introduced the metamodel of the infrastructure layer and have presented the graphical notation of the conceptual context model and an XML-based notation of the operational context model. We have implemented a context server based on the three-layered context metamodel. Using the context server we are able to implement a concrete context model by description and to build the context infrastructure of the smart home environment without further programming.

Currently the transition between the conceptual model and the implementation-independent representation of the operative context model has to be done manually. We will investigate the usage of domain specific language tools for a semi-automatic transition. Another aspect for future work is the definition of a reference context model on the infrastructure layer, towards realizing an AAL service marketplace.

References

[1] S. Meyer, A. Rakotonirainy, ”A Survey of Research on Context-Aware Homes”, Proc. of the Australasian information Security Workshop Conference on ACSW Frontiers 2003, vol. 21, pp. 159-168, 2003.

[2] OSGi Alliance, Osgi Service Platform, Release 3, IOS Press Inc, 2003.

[3] M. Weiser, R. Gold, J. S. Brown, The origins of ubiquitous computing research at PARC in the late 1980s, IBM Systems Journal vol. 38(4), pp. 693-696, 1999.

[4] M. H. Coen, ”Design Principles for Intelligent Environments”, Proc. of the Fifteenth National Conference on Artificial Intelligence (AAAI’98), pp. 547-554, 1998.

[5] J. Meis, J. Draeger, ”Modeling automated service orchestration for IT-based homeservices”, Proc. of the IEEE/INFORMS Int. Conference on Service Operations and Logistics and Informatics SOLI ‘07, pp. 155-160, 2007.

[6] A. Dey, “Providing Architectural Support for Building Context-Aware Applications,” PhD Thesis, Georgia Institute of Technology, 2000.

[7] B. Schilit, “System architecture for context-aware mobile computing”, unpublished doctoral dissertation, Columbia University, New York, 1995.

[8] C. Becker, D. Nicklas, “Where do spatial context-models end and where do ontologies start? A proposal of a combined approach”, Proc. of the First Int. Workshop on Advanced Context Modeling, Reasoning and Management, 2004.

[9] T. Strang, C. Linnhoff-Popien, “A Context Modeling Survey”, Proc. of the First Int. Workshop on Advanced Context Modeling, Reasoning and Management, 2004.

[10] M. Samulowitz, F. Michahelles, C. Linnhoff-Popien, “An architecture for context-aware selection and execution of services.”, New developments in distributed applications and interoperable systems, Kluwer Academic Punlishers, pp. 23-39, 2001.
[11] A. Held, S. Buchholz, A. Schill, “Modeling of context information for pervasive computing applications”, Proc. of SCI/ISAS, 2002.
[12] F. Pagnanelli, D. Giuli, “An Ontology-Based Context Model for Home Health Monitoring and Alerting in Chronic Patient Care Networks”, AINAW Advanced Information Networking an Applications Workshop, 2007.
[13] T. Gu, X. H. Wang, H. K. Pung, D. Q. Zhang, “Ontology Based Context Modeling and Reasoning Using OWL”, Proc. of the 2004 Communication Networks and Distributed Systems Modeling and Simulation Conference, 2004.
[14] H. Chen, T. Finin, A. Joshi, “An Ontology for Context-Aware Pervasive Computing Environments”, The Knowledge Engineering Review, Cambridge University Press, pp. 197-207, 2003.
[15] K. Henricksen, J. Indulska, A. Rakotonirainy, ”Modeling Context Information in Pervasive Computing Systems”, Proc. of the Int. Conference on Pervasive Computing, LNCS 2414, pp. 167-180, 2002.
[16] K. Henricksen, S. Livingstone, J. Indulska, ”Towards a hybrid approach to context modelling, reeasoning and interoperation”, Proc. of the First Int. Workshop on Advanced Context Modelling, Reasoning and Management, 2004.
[17] D. Preuveneers, J. Van den Bergh, D. Wagelaar, A. Georges, P. Rigole, T. Clerckx, Y. Berbers, K. Coninx, V. Jonckers, K. De Bosschere, “Towards an extensible context ontology for Ambient Intelligence”, Ambient Intelligence: Second European Symposium, EUSAI, 2004.
[18] J. Nehmer, M. Becker, A. Karshmer, R. Lamm, “Living assistance systems: an ambient intelligence approach.”, Proc. of the 28th Int. Conference on Software Engineering, 2006.
[19] M. Baldauf, S. Dustdar, F. Rosenberg, „A survey on context-aware systems“, Int. Journal on Ad Hoc and Ubiquitous Computing, Vol. 2, No. 4, 2007.
[20] P. Hu, J. Indulska, R. Robinson, ”An Autonomic Context Management System for Pervasive Computing”, Proc. of the Sixth Int. Conference on Pervasive Computing and Communications, 2008.
[21] M. Anastasopoulos, C. Bartelt, J. Koch, D. Niebuhr, A. Rausch, “Towards a Reference Middleware Architecture for Ambient Intelligence Systems”, Proc. of the Workshop for Building Software for Pervasive Computing, 20th Conference on Object-Oriented Programming Systems, Languages and Applications, 2005.
[22] T. Kleinberger, M. Becker, E. Ras, A. Holzinger, P. Müller, “Ambient Intelligence in Assisted Living: Enable Elderly People to Handle Future Interfaces”, Universal Access in HCI, LNCS 4555, pp. 103–112, 2007.
[23] M. Wojciechowski, J. Xiong, “A user interface level context model for ambient assisted living”, Proc. of the Int. Conference on Smart Homes and Health Telematics, LNCS 5120, pp. 105-112, 2008.
[24] M. Wojciechowski, “End User Context Modeling in Ambient Assisted Living”, Int. Journal of Advanced Pervasive and Ubiquitous Computing, in press.
[25] S. Yau, J. Liu, “Hierachical Situation Modeling and Reasoning for Pervasive Computing”, Proc. of the IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, pp. 5-10, 2006.