Experiences from Model-Driven Development of Homecare Services: UML Profiles and Domain Models

Ståle Walderhaug[1,2], Erlend Stav[1], and Marius Mikalsen[1]

[1] SINTEF ICT, SP Andersens vei 15b, N-7465 Trondheim, Norway
[2] Department of Computer Science, University of Tromsø, N-9037 Tromsø, Norway
{stale.walderhaug,erlend.stav,marius.mikalsen}@sintef.no

Abstract. Model-driven development approaches such as Model Driven Architecture (MDA) have been proposed as the new paradigm for software development. The adoption of MDA is still low, partly because of the general-purpose modelling language being used. Domain specific modelling languages are being developed for technological and industrial domains to improve the expressiveness and effect of model-driven development techniques. The healthcare domain could benefit from these methodologies. In order to incorporate domain knowledge in a MDA process, information about workflows, artefacts and actors can be formalized in a UML profile and applied by MDA tools for design and development. This paper presents the work done on model-driven development of smart homecare services in the MPOWER project. Following an iterative approach, two UML profiles to support development of Service Oriented Architecture based homecare applications are proposed. Using homecare specific UML profiles indicate an improvement in the process for model-driven development of homecare services.

1 Introduction

Model-Driven Development (MDD) such as OMG's Model Driven Architecture (MDA) [1], has the potential to improve the quality of software systems. Quality attributes such as interoperability, reusability and appropriateness of software components and systems are main features of MDA. By using abstraction and advanced automation techniques, software artefacts are created from formal models that are represented using languages such as the Unified Modeling Language (UML) [2]. The core of the MDA process, and similar MDD processes, is to use formal models as the main development artefacts in the entire development process, from domain analysis to implementation, deployment and testing.

Domain specific modelling languages (DSML) have been proposed as a means to overcome many of the shortcomings with UML and MDA. The scientific knowledge about applying MDD techniques in design and development of healthcare information systems is scarce [3]. Creating DSMLs for the healthcare domain is a daunting task, and requires extensive investment of resources and time.

M.R.V. Chaudron (Ed.): MODELS 2008 Workshops, LNCS 5421, pp. 199–212, 2009.
© Springer-Verlag Berlin Heidelberg 2009

We set out to investigate how MDD with DSML support should be introduced and applied in a healthcare sub domain. In the MPOWER project [4], we have developed a framework for creating homecare software services using a model-driven approach. The framework defines a MDA toolchain which is a set of modelling, transformation and development tools that supports the complete MDA process as descibed in the MDA Guide [1]. A comprehensive model of actors and services in homecare along with the MDA toolchain for designing and implementing these domain specific web services has been developed and evaluated. This paper presents research results from the project with focus on:

1. What is the domain knowledge in homecare that can be used as assets in the MDA process?
2. Which knowledge can be included in a UML Profile for homecare services and how can this knowledge be utilized by developers?

The MPOWER toolchain, providing model traceability, model transformation and code generation, has been evaluated in the development of two proof-of-concept applications and is currently been redesigned with improved UML Profile support for the domain. Based on the experience from developing the MPOWER framework and proof-of-concept applications a conceptual domain model and UML profile for service oriented computing in the homecare domain is proposed and discussed.

The remainder of this paper is organized as follows. The next section describes the background for the work, including relations to and motivations for applying model driven development, domain specific modelling languages, and service oriented architecture. Then the applied method and main activities within the MPOWER project are described. The main results from each of the main activities are presented next, including conceptual domain models and our preliminary DSML approach based on two UML profiles. A discussion follows this, before we conclude the paper.

2 Background and Related Work

The work presented herein is a part of the EU-IST project MPOWER (contract no. 034707) and of an ongoing PhD thesis work by the main author. MPOWER is a user driven research and development project where the main goal is to create a framework for rapidly creating standards-based homecare services. The framework includes the definition of a toolchain which is being used in the development of two proof-of-concept applications targeting elderly and cognitive impaired people living at home.

MDD promises a potential to improve the quality of software systems and their development by using formal models as first class entities in the entire development process. When MDD is done properly, improvements in the design, development and maintenance processes can be achieved, in addition to improving the interoperability aspects and reducing the overall development costs.

Tuomainen et al argues that modelling helps the understanding of healthcare activities by being illustrative, identifying improvements, simulate organisational

processes and individual activities in healthcare [5]. They compare three model centric approaches; MDA, Business Process Modelling with BPMN and BPEL and the HL7 development framework. They conclude that in order to realise their full potential these approaches require local and project specific adaptation. This paper explains such an adaptation for the homecare domain.

3 Methods

The main objective of the MPOWER project is to create a framework that facilitates rapid development of homecare services. To achieve this, it is imperative to acquire knowledge about the homecare domain, and make this domain knowledge available to actors involved in the system development processes. Due to the complexity of the healthcare domain, it was considered necessary to iterate between domain modelling and system design. To facilitate this interaction, the MPOWER project defined three main activities:

1. Capture domain knowledge from experts on aging/dementia, healthcare workers in the domain, family carers and patients.
2. Specify a MDA toolchain that support documentation of system requirements, modelling of design and development of services. Moreover, the toolchain must be evaluated in terms of usability and usefulness/performance by implementing two Proof-of-Concept Applications (PoCA). The results are a MDA toolchain with evaluation reports on developer acceptance and technical qualities
3. Design a DSML that incorporates the domain knowledge from task 1 and MDA toolchain experience from task 2. The result will be one or more UML profiles that can be used with a revised MDA toolchain.

3.1 Activity 1: Capture Domain Knowledge

The MPOWER project focuses on smart homecare solutions for elderly and cognitive impaired people. The domain models for the work being presented in this paper can be seen from two different viewpoints:

1. The Homecare viewpoint: this viewpoint focuses on organizational aspects of homecare as well as the main stakeholders (people and systems) involved. This model is the result of a comprehensive process involving a total of 140 domain stakeholders such as domain experts, professional caregivers, family carers and patients [6].
2. The Homecare SOA viewpoint: this viewpoint focuses on the main system components and their relationships in terms of the principles of Service Oriented Architecture design. Important assets for this model are the design principles given by Erl [7], and SOA4HL7 methodology [8]. The structure and semantics of the domain model is supported by the SOA reference architecture from IBM along with the IBM UML profile for software services [9].

3.2 Activity 2: Designing a Toolchain for MDD in Homecare

To have a formal way of specifying the domain models, proper modelling tools are needed. In the beginning of the MPOWER project, a set of tools were specified as the MPOWER toolchain to be used by all involved personnel for conceptual modelling, requirements specification, analysis, system design, system development, deployment and testing. The process used for selection of tools matches the recommendation given by Staron [10], page 240: "The process [of creating domain models] should be tool independent. The independence should be supported by using technologies that are open and unbounded, but at the same time supported by more than one tool." In MPOWER, Microsoft Word 2003 was used for describing user scenarios and Enterprise Architect (EA) V6.5 from Sparx Systems used for UML modelling of use cases and services, model transformation and code generation of WSDL code. THE IBM's UML 2.0 profile for Software Services was applied during modelling the services to structure models and stereotype coreelements. Available from IBM [9]. Finally, NetBeans V6.0 was used as Java IDE for generating service skeletons and implementing the services. `http://www.netbeans.org`

The described toolchain was used from the start of the project with only minor modifications such as EA upgrades and bug fixes. The two PoCAs were developed using the toolchain and the performance of the toolchain, were investigated from two perspectives: 1) Developer acceptance of the MPOWER toolchain: using the Technology Acceptance Model with two additional factors, as reported in [11], and 2) Technical review: Weekly scrum and quarterly technical meetings with workshop sessions on how to improve the toolchain.

3.3 Activity 3: Refine the MPOWER Toolchain and Develop a DSML

The UML standard allows for the creation of a DSML in two ways: 1) Creating a new language based on Meta Object Facility [12], or 2) extending UML through the use of UML Profiles. As discussed by Selic, the latter will often be the most practical and cost-effective solution [13], and is also the chosen method for this paper. By using UML profiles to create a DSML, the semantics and syntax of UML can be inherited, and powerful UML/MDA tools can be used with the profile for software development.

The UML Profile standard [2], outlines several reasons for creating a DSML from UML. The most pertinent reasons for the challenges addressed in this paper are:

- Terminology adapted to the healthcare domain
- Add information that can be used during transformation
- Add constraints that restrict the way you can use the metamodel

There is not much knowledge in the scientific community about best practices for creating DSMLs with UML [13] [14]. In a paper from 2007, Selic summarized the basic steps for creating a DSML in terms of UML in [13]: 1) Create a

conceptual domain model that includes the essential concepts of the domain, the relationships between the concepts, the constraints that govern the use of the concepts. A selection of UML models from Activity 1 makes up the conceptual domain model, 2) Map domain model to a UML profile, refining the core UML specification with stereotypes, tagged values and constraints.

The process of creating a domain specific UML profile is not straightforward, since the level of abstraction and the intended use of the profile play an important role for the definition of the profile elements. This challenge is tackled with experience from the design of the MPOWER toolchain and development of two MPOWER PoCAs for the homecare domain. To identify and model the elements of a UML profile is an iterative process. To guide this process, the Staron's guidelines for defining good stereotypes using a classification schema [10], is used.

4 Results

The results presented in this section are based on the work carried out in the MPOWER project from October 2006 to June 2008.

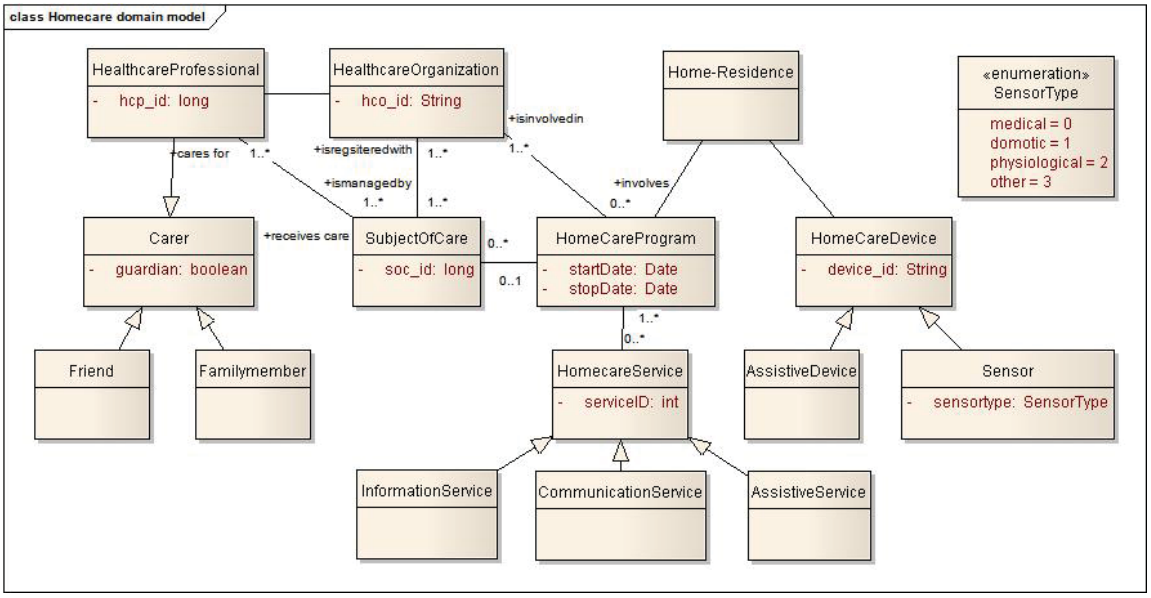

Fig. 1. Diagram showing the main concepts in a smart homecare domain

4.1 Activity 1: Conceptual Domain Models

The domain models were developed in several iterations from October 2006 to September 2007. Figure 1 shows the main concepts from a homecare viewpoint.

To keep the model at an abstract level and not overpopulate it with unnecessary details, most attributes on the classes are hidden. The main classes and relationships are:

Table 1. The main classes and relationships in the homecare domain

Sterotype	Comment
Subject of Care (SoC)	person receiving care through a homecare program. The SoC has a unique identifier that is managed by the assigned healthcare organization. A SoC must be associated with at least one healthcare professional
Homecare Program	a class comprising the services, devices and healthcare organizations involved in providing homecare service to a SoC.
Carer	an individual that is a part of the family, a healthcare professional or a friend. All HealthcareProfessionals must be associated with a HealthcareOrganization.
Healthcare Organization	an organisation that is directly involved in the provision of care to a SoC.
Homecare Service	a service provided to the SoC through a Homecare program. Three core types of services have been identified: information service (e.g., calendar, medication list), communication services (e.g. SMS, email), and assistive service (e.g. indoor location service, heating control, burglar alarm, oven control).

Concepts in the model are aligned with the concepts presented in Continuity of Care (CONTSYS) standard [15], and service categories from [16]. Most concepts are also available in the HL7 RIM, but CONTSYS is more specific than HL7. These resources were found useful in selecting an appropriate abstraction level and structure in the domain model. The complete models of actors and services are presented in [6]. Figure 2 shows the main components, stereotyped with the five layers of the IBM SOA reference architecture.

4.2 Activity 2: The MPOWER Toolchain

The experiences from using the described toolchain for development of the services that form the PoCAs are grouped into developers' subjective experience and technical experience. The first group entails perceived characteristics such as the factors described by the Technology Acceptance Model [17]. The results from a developer evaluation of the MPOWER toolchain is presented in [11], and concludes that perceived ease of use and perceived usefulness are factors that affect the developers' adoption of MDA. It was also found that traceability between artefacts in the development process was useful. A major drawback with the evaluated toolchain was found to be the incomplete code generation features. A technical review of the MPOWER toolchain revealed that 1) WSDL model transformation incomplete: it was necessary to customize the transformation template for WSDL models, 2) WSDL code generation had errors: the built-in transformations in the EA tool generated some errors that had to be changed manually, e.g. using "type" references instead of "element" references in message definitions, and 3) Performance problems using HL7: the import of HL7 message types into WSDL resulted in tool crashes because of memory allocation

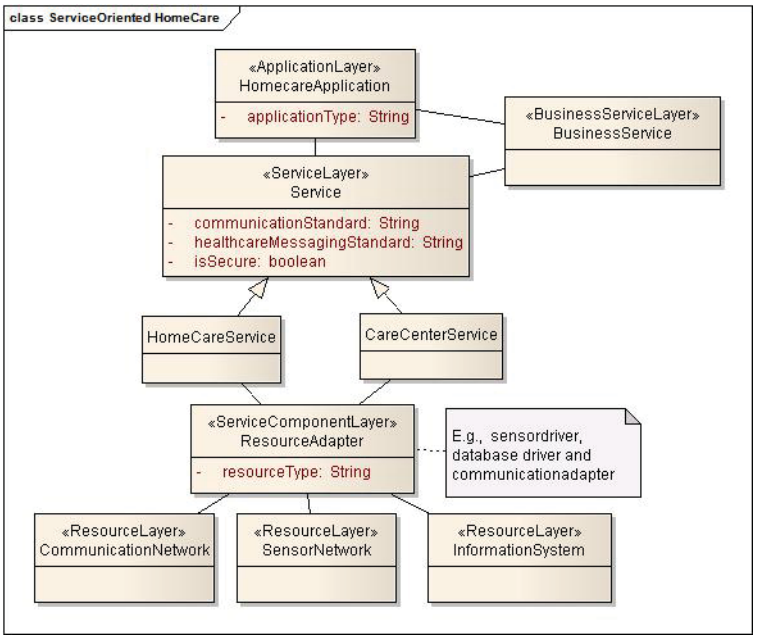

Fig. 2. The Service-oriented view on a typical homecare environment

problems. Recursive import of HL7 xml schema (xsd) defintions were not handled by the WSImport tools in Netbeans.

From the experience with the MPOWER toolchain, a set of new features were proposed. The developers would like more support in generating the implementation of the services – repetitive code (e.g. for DB management, handling of security, return status), and support for object/relational persistence service, such as generation of Hibernate mappings for the information elements declared in the message definitions of the WSDLs. These features may impact the design of a DSML as they could require domain specific information to be incorporated into the models during the service design.

4.3 Activity 3: Refined Toolchain - Mapping of Domain Concepts to DSML - UML Profile

The process of defining a UML Profile for SOA in homecare use concepts from the conceptual domain models and experience from toolchain and PoCA development in an iterative approach. This section presents the preliminary results from Activity 3 after the first iteration (January-June 2008). Activity 3 is carried out by a core team of three researchers. The profiles were updated in three main revisions: initial version for the start of the development, second version after first version of services and the third version after the first iteration of the PoCA development. The changes between version one and two were significant, whereas only tags and minor adjustments to relationships were done for version 3.

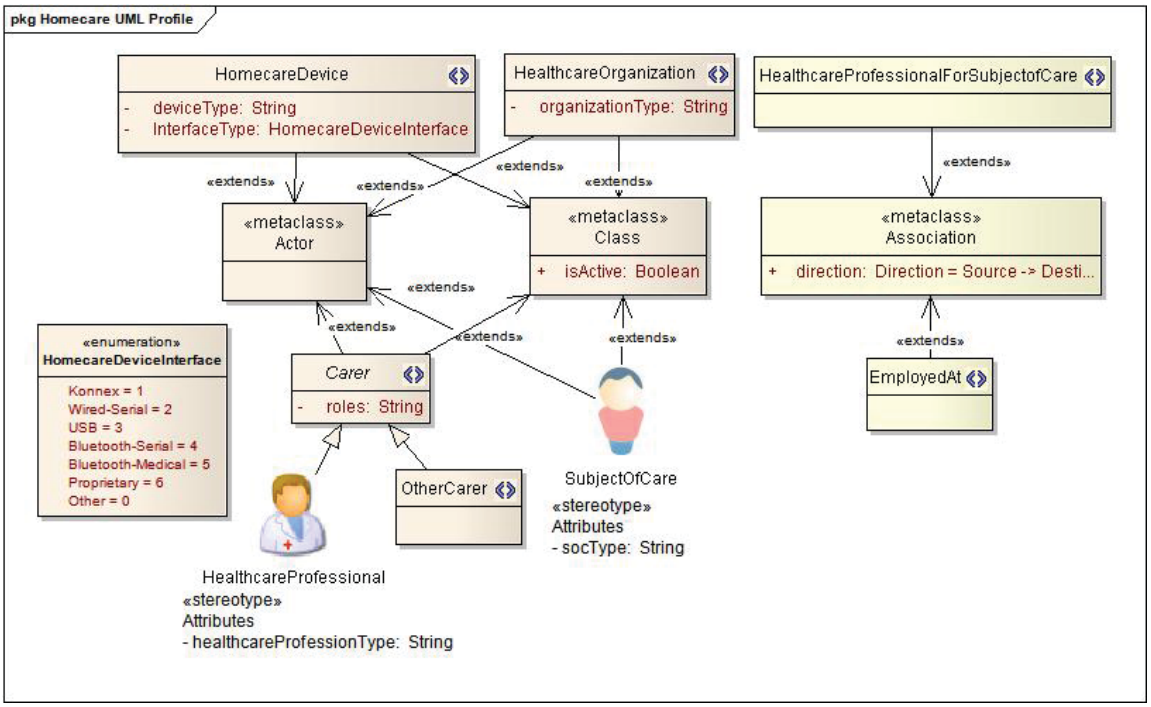

Fig. 3. First version of Homecare UML Profile

Two UML profiles are proposed, the Homecare UML Profile and the SOA Homecare UML Profile. The profiles can be used separately or together in a MDA development project, depending on how the profile elements are utilized in the development process.

Figure 3 shows the Homecare UML profile. The profile elements are mainly derived from the homecare conceptual model (figure 1). The tagged values on the stereotypes were identified based on experience from the MDA toolchain work. The mapping of concepts to the UML profile was also guided by the CONTSYS standard [15].

All the stereotypes in the Homecare UML Profile falls into the category "Virtual Metamodel Extension, restrictive" defined by Staron in [10]. These are stereotypes that reuse the semantics of the metaclasses (e.g. Actor and Class). Often they must be used with other stereotypes, making the stereotyped model element more precise and may also add a new icon to the concrete syntax to familiarize the model presentation (e.g. icons on HealthcareProfessional and SubjectOfCare).

The SOA Homecare UML profile enables developers to create precise models of SOA-based homecare systems. Figure 4 shows the core elements of the SOA Homecare UML profile.

Table 2. Table describing the proposed stereotypes in and tagged values in the Homecare UML profile

Sterotype	Comment
Subject Of Care	Subject of care (SoC) is defined in CONTSYS as "person seeking to receive, receiving, or having received health care" [15]. Used to decorate SoC modelling elements and to add information about the SoC that can be used during model transformation or code generation. SoC_type: describes can be used to describe different types of SoC according to e.g., national specific patient classifications.
Carer	A stereotype that should be used on all modelling elements representing an individual that provides care, professionals as well as non-professional caregivers. roles: can be used to set the default role, e.g. in terms of security, for the instances of classes marked with this stereotype
Healthcare Professional (HcP)	Defined in CONTSYS as "person authorised by a nationally defined mechanism to be involved in the direct provision of certain health care activities" [15]. Should be used to mark all modelling elements of type Class/Actor that are representing individuals that fit this definition. The roles attribute is inherited from Carer.
Other Carer	Defined in CONTSYS as "person providing assistance for activities of daily living or social support". This stereotype should be used to mark modelling elements of type Class/Actor that are representing individuals such as family members, friends and other carers employed by non-healthcare organizations such as home services and security services. The roles attribute is inherited from Carer
Healthcare Organization (HcO)	Defined in CONTSYS as "organisation involved in the direct provision of health care" [15]. This stereotype should be used to mark all modelling elements of type Class/Actor that represents organisations that fits the CONTSYS definition. orgainsationType: is used to describe the type of organisation according to speciality levels, private versus public or other national classifications.
Homecare Device	Generic homecare device stereotype to be used on modelling elements of type Class/Actor. The stereotype can be useful for design-time checking of interoperability and interconnectivity of devices in a homecare system. deviceType: describes the type of this device interfaceType: describes the kind of interface used to connect to this device. In the UML profile, the HomecareDeviceInterface enumeration is defined based on the experience in the MPOWER project, but can be refined to fit other technologies.

Table 2. (*continued*)

Sterotype	Comment
Healthcare Professional For Subject Of Care	A stereotype that is used to mark an association between a HealthcareProfessional and a SubjectOfCare. The stereotype can be used to ensure that a SubjectOfCare is associated with at least one HealthcareProfessional.
EmployedAt	A stereotype that is used to mark an association between a HealthcareProfessional and a HealthcareOrganisation. The stereotype can be used to check that all HealthcareProfessional "types" are employed at a HealthcareOrganisation "type".

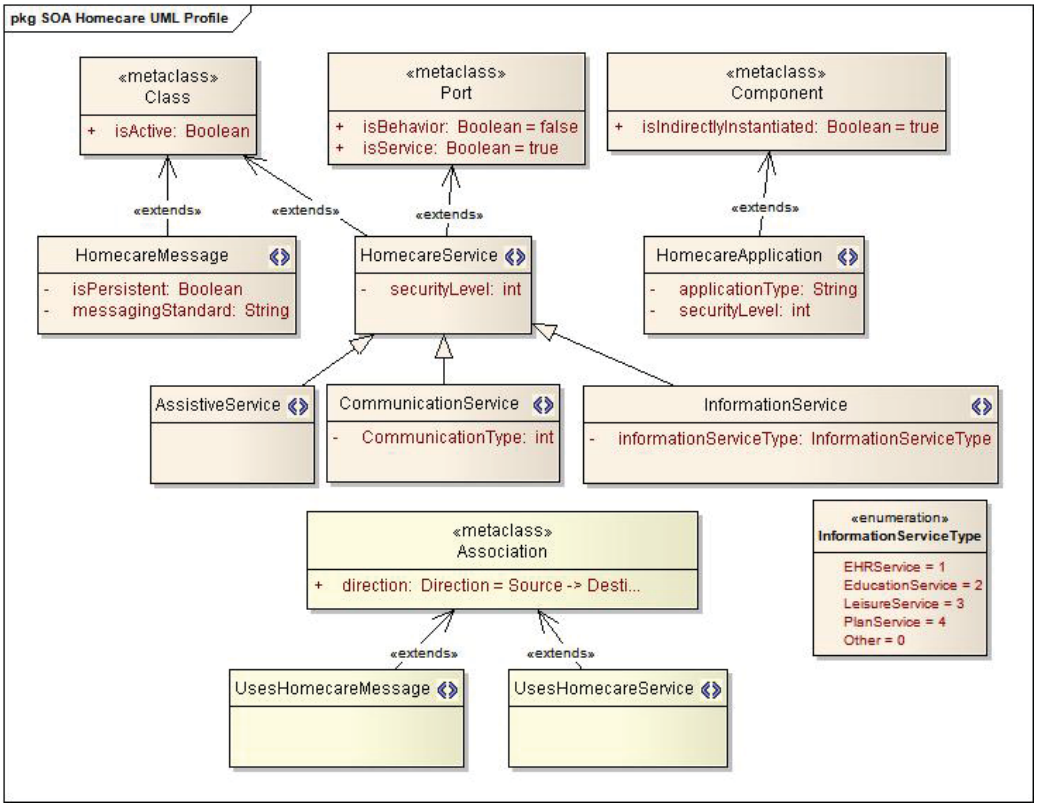

Fig. 4. The Homecare SOA UML Profile diagram

All stereotypes in the Homecare SOA UML profiles falls into the category "Code generation, restrictive" defined by Staron [10]. These are stereotypes that extend the base metaclass (e.g., Class and Port) with some properties to increase the precision of the semantics and restrict the usage. Each stereotype is described in more detail in the table below.

Table 3. List of stereotypes and tagged values in the SOA Homecare UML profile

Sterotype	Comment
Homecare Application	The stereotype adds two properties to the modelling elements: securityLevel: this describes the security level of the application. This can be used to check that a user (i.e., service, component, application) of the application must have at least the same access level in order to be allowed to use the service. applicationType: this describe the type of application this is, e.g. in terms of deployment configurations.
Homecare Service	A service which is used in the homecare environment. The IBM Service Profile should be used in combination with this stereotype. securityLevel: this describes the security level of the service. This allows for specification of the security requirements and rights for a modelling element that can be utilized during code generation. In the next version of the UML Profile, this stereotype will be updated with more security tags addressing service-service authorization and information encryption.
Homecare Message	To denote elements that are messages used in interactions of homecare services and applications. isPersistent: indicates whether the message data is stored in a database or not. This can be used for creating Hibernate mapping code and database schema. messagingStandard: the standard to which this message belongs. Can be used both for code generation, ensuring correct libraries are present, and for checking conformance with the standard.
Assistive Service	A type of homecare service that provides assistive functionality in the homecare system. Derived from Stefanov's classification for smart house services for elderly and cognitive disabled [16].
Information Service	An information service which will be used by stakeholders in a homecare setting. serviceType: defines the type of information service this service belongs to. An enumeration is proposed based on the experience in the MPOWER project.
Uses Homecare Message	A stereotype that mark associations between a HomecareService and a HomecareMessage. The stereotype can be used to generate traceability information that can again be used when messaging standards are being updated or changed.
Uses Homecare Service	A stereotype that mark associations between a HomecareApplication and a HomecareService. The stereotype can be used to generate traceability information that again can be used when a homecare service is being updated or changed.

5 Discussion

The work presented in this article is a part of an initiative to develop a MDD framework for healthcare, focusing on homecare services in the first phase. It is considered imperative to incorporate domain knowledge into the framework and make this knowledge readily available for architects and developers in all development phases. This paper presents the results from creating a domain specific modelling language for homecare using UML Profiles.

Capturing the conceptual domain knowledge for homecare, or any other healthcare sub-domain, is a daunting task. Many stakeholders are involved, as well as a plethora of information systems, involving many different coding standards and vocabularies. These factors, in addition to legislative factors and organisational aspects, make modelling of reusable healthcare domain models difficult, but not impossible. To succeed in creating a useful model-driven software development process, it is important to choose the right level of abstraction and in divide the healthcare application areas into well defined sub-domains.

The homecare domain model shown in figure 1 shows the most important actors and relationships between them. The model would fit for modelling most homecare solutions, is aligned with the CONTSYS standard [15], and includes the main classifications from Stefanov's paper on smart house technologies for elderly [16]. The model is the result of a comprehensive domain analysis process where 140 domain stakeholders from four European countries were involved in improving the validity of the model [6]. If new concepts are developed for the domain, these can be added as an extension to the existing homecare model, without compromising the original model and the related UML profiles.

The Homecare SOA model is based on the domain investigation from the MPOWER project, in addition to the IBM SOA reference model and IBM UML profile for software services [9]. The Homecare SOA model provides information about deployment of services and possible configuration and information sources. The model is on an abstract level, and could in certain cases be refined with details about security platforms and network connectivity details. Such domain knowledge could be useful in planning the distribution of services and integration with existing resources, but will also make the model less suitable for reuse across different healthcare enterprises and nations.

To make the domain knowledge readily available as assets in the development process, UML profiles were chosen, inline with the recommendations by Selic [13]. UML profiles builds upon the syntax and semantics of UML, and most UML tools support profiles. This is an imperative advantage, enabling developers to use their favourite UML tool for design and development. The process of selecting domain concepts to include as stereotypes, tagged values or constraints in a UML model, requires knowledge about model-driven development, but also experience from modelling systems in the domain in question. Experience from the development of a MDA toolchain (Activity 2) provided information about which target software artefacts that should be generated from the models and which models and diagrams that should be applied for achieving this. This information was of utmost importance when choosing the metaclass extensions for the elements in the UML profiles.

UML profiles are used to customize the modelling language to include familiar concepts that enables more effective and precise system design and implementation. Stereotypes can be used for several purposes, as discussed by Staron [10]. The result from the mapping of domain concepts to a DSML (Activity 3) showed that all stereotypes in the Homecare UML profile are classified as Virtual Metamodel Extensions. This implies that this profile is mainly used to increase the expressiveness of the modelling language when designing systems for homecare. A "virtual metamodel extension, restrictive" stereotype adds a domain specific icon such as a picture of a nurse to the modelling element, together with a well known domain specific label such as HealthcareProfessional.

The Homecare SOA UML Profile includes elements from the "Code generation, restrictive" category. These stereotypes can improve code generation by providing domain information so that code generation scripts can create high-quality code. In this paper code generation was restricted to WSDL and Hibernate code, but other software artefacts can be generated from the domain information in a UML profile. Test cases, error-checking code, security policy verification, and result validation may also be generated.

The two proposed UML profiles can be used on the same models to provide different "views". In addition, the IBM Software Service UML profile should be used to complement the service design models for SOA Homecare systems.

6 Concluding Remarks

Model-driven development approaches can be improved by extending the modelling language with domain specific concepts. UML Profiles can be used as a mechanism for toolchains based on OMG's MDA and UML standards.

The UML Profiles must provide information that can improve the design and/or code generation processes. The two profiles proposed in this paper are based on solid work on capturing homecare domain knowledge and experience from developing homecare SOA systems using MDA. Though the profiles are still undergoing updates and improvements, they can improve modelling process performance and results.

It was found useful to have a development activity in parallel with specification of the conceptual domain and profiles. The experience from the development activity gave valuable input to the mapping of concepts to DSML processes. This finding extends the approaches to DSML development in [13] [14] [18].

The MPOWER Toolchain will be evaluated by university students in 2008. In 2009, an experiement measuring the subjective improvements (perceived characteristics) and objective improvements (e.g., reduction of errors, time spent for development) will be conducted with 20 developers from the healthcare domain.

References

1. Miller, J., Mukerji, J.: MDA Guide Version 1.0.1. In: Miller, J., Mukerji, J. (eds.) Object Management Group (OMG), pp. 1–62 (2003)

2. Object Management Group (OMG), UML 2.1.2 Superstructure and Infrastructure, Object Management Group (OMG) (2007)
3. Mohagheghi, P., Dehlen, V.: Where Is the Proof?-A Review of Experiences from Applying MDE in Industry. In: Schieferdecker, I., Hartman, A. (eds.) ECMDA-FA 2008. LNCS, vol. 5095, pp. 432–443. Springer, Heidelberg (2008)
4. MPOWER Consortium (2006) (cited November 15, 2006), http://www.sintef.no/mpower
5. Tuomainen, M., et al.: Model-centric approaches for the development of health information systems. In: Medinfo. 2007 Brisbane, Australia (2007)
6. Walderhaug, S., Stav, E., Mikalsen, M.: Reusing models of actors and services in smart homecare to improve sustainability. In: Medical Informatics Europe 2008, Gothenburg, Sweden. IOS Press, Amsterdam (2008)
7. Erl, T.: Service-Oriented Architecture Concepts. In: Erl, T. (ed.) Technology, and Design, Crawfordswille, Indiana, USA. The Prentice Hall Service-Oriented Computing Series. Prentice Hall, Englewood Cliffs (2006)
8. Honey, A., Lund, B.: Service Oriented Architecture and HL7 v3: Methodology, HL7 Service Oriented Architecture Special Interest Group (SOA SIG). p. 79 (2006)
9. Johnston, S.: UML 2.0 Profile for Software Services (2005) (cited November 15, 2008), http://www.ibm.com/developerworks/rational/library/05/419_soa/
10. Staron, M.: Improving modeling with UML by stereotype-based language customization. In: School of Engineering, p. 270. Blekinge Institute of Technology, Blekinge (2005)
11. Walderhaug, S., et al.: Factors affecting developers' use of MDSD in the Healthcare Domain: Evaluation from the MPOWER Project. In: From code-centric to model-centric develpoment, Workshop at European Conference on Model-Driven Architecture. European Software Instituite, Berlin (2008)
12. Object Management Group (OMG), MOF 2.0 / XMI Mapping Specification, v2.1 (2005)
13. Selic, B.: A Systematic Approach to Domain-Specific Language Design Using UML. In: 10th IEEE ISORC (2007)
14. Lagarde, F., et al.: Improving uml profile design practices by leveraging conceptual domain models. In: Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering, pp. 445–448 (2007)
15. CEN TC251, EN 13940-1: Health Informatics - System of Concepts to Support Continuity of Care - Part 1: Basic Consepts. European Committee for Standardization, p. 105 (2006)
16. Stefanov, D.H., Bien, Z., Bang, W.-C.: The smart house for older persons and persons with physical disabilities: structure, technology arrangements, and perspectives. IEEE transactions on neural systems and rehabilitation engineering 12(2), 228–250 (2004)
17. Davis, F.D., Bagozzi, R.P., Warshaw, P.R.: User Acceptance of Computer Technology: A Comparison of Two Theoretical Models. Management Science 35(8), 982–1003 (1989)
18. Fuentes-Fernández, L., Vallecillo-Moreno, A.: An Introduction to UML Profiles. UML and Model Engineering 5(1) (2004)