Disability centered approach in smart space management

Rachid Kadouche[1], Bessam Abdulrazak[1], Sylvain Giroux[1], and Mounir Mokhtari[2]
[1] *DOMUS Lab Université de Sherbrooke, Sherbrooke, Québec, Canada*
[2] *Handicom Lab, Institut National des Télécommunications-GET, Evry, France*
Rachid.kadouche@usherbrooke.ca

Abstract

The recent convergence of ubiquitous computing and context-aware computing has seen a considerable rise in interest in applications that exploit aspects of the contextual environment to enhance implicit user interaction, offer services, present information, tailor application behavior or trigger adaptation. However they are often insufficient to provide adequate environments for people with special needs and allow them to be an integral part of society. This paper presents our approach to enhance environment services to peoples with special needs. Particularly it highlights the data representation and implementation of this approach under a framework based on the Web Ontology Language OWL-DL. A prototype of this framework was integrated in a smart home demonstrator and tested with users under a laboratory conditions.

Keywords: *Semantic Matching Framework, Smart Space Management*

1. Introduction

Research in Smart Home is becoming an important trend [1-2] [6-7]. It is at the crossroad of research in pervasive computing, ambient intelligence, machine learning, databases, mobile computing, robotics, and multimedia. In general, the aim is to maximize comfort and productivity of the inhabitants while minimizing costs (installation, operation and maintenance). At the same time, there are huge needs in healthcare to maintain at home dependent people. We are convinced that smart spaces and smart homes can dramatically improve the autonomy of people with disabilities in real-life habitats with assistance and remediation through technology. However dependent people have a great variability in their needs and levels of motor and/or cognitive handicap. Hence they call for personalized services, especially when interacting with their environment. In smart homes, the Current solutions are often insufficient and costly as they require the help of qualified personnel involving several disciplines (ergonomics, occupational therapy, design, engineering, medicine, etc.). One of the big challenges is to understand the interaction between the human and his environment before designing ubiquitous assistive environment dedicated to people with special needs. This leads to study the disability and its impact on the environment.

According to Fougerollas [3], disability is not only a physical deficiency, but it results mainly from the inadequacy of the environment infrastructure to fulfil the user needs. This defines the "***handicap situation***" which indicates that, disability doesn't design the individual, but mainly represents the inadequacies of the living environment in regard to the user's capabilities. The main key is how to identify the differences between a person having severe disabilities and a one without a disability in terms of daily activities? Or even which deficiency of the user is causing the handicap situation and what are the environmental elements becoming an obstacle for the user? And, how can ubiquitous environment compensate, partly or totally, disability of each person. Each of these questions is strongly dependent on human and environment factors that should be supported by any assistive system. Our approach, to adapt the environment to people with special needs, is based on the detection of the handicap situations in the user's environment which gives as global vision on the user limitation capabilities (the accessibility rate of each user) and provides adapted process to personalize the service delivery. To achieve that, we focus on the interaction between the human characteristics and its environment which we have defined as Human Environment Interaction (HEI).

2. Related Work

Many system architectures are specifically addressing support for assisting people with special needs in providing both effective and economical care in the home either by building an infrastructure equipped with sensors, panels, cameras, etc. integrating an observation rooms for monitoring the interaction and activities of the elderly within the environment (exp: [4-9]), or focusing on more specific aspects of the use of technologies to improve living conditions, some of them are focussing on a specific disability exp: DOMUS LAB [10] for helping peoples with dementia to carry out a daily activity task; the Assisted Cognition Project [11], provides techniques to enhance the quality of people suffering from Alzheimer disease. Intel's Proactive Health Research [12], explores the different ways to support the daily health and wellness living.

Most of these systems have progressed in various aspects of pervasive computing, but are weak in supporting knowledge sharing and context reasoning due to their lack on common ontology with explicit semantic representation. In addition, they mainly focus on monitoring and usually cater to a specific application and sensor suite. Furthermore, none provides generic system that dynamically explores the environment to identify the different barriers which prevent the user to access to some environment services and deliver the user limitation capabilities or handicap situations in each environment.

Then, there is PPH model [3] that treated also the interaction problem between the environment and the human and defined the handicap as an inadequacy between the environment factor and the human factor, and the ICIDH model [13] which is a conceptual framework based on a cause-and-effect relationship between impairment and handicap. Both projects are conceptual approach, they are based on sociological parameters related to the users and the environment; they are limited to a specific applications (social and professional integration, etc.) and used by particular community such as sociologist, ergonomist. Our work is similar to those last two projects they are also based on defining the handicap situation in an environment. But they are different from our task of presenting a technical model, which could be handled by any technological system, based on clinical, sociological, and usage analysis studies in the field of assistive technologies. We also designed the users and environment characteristics and quantified the relationship between the user's physical parameters and the technical parameters of the environment under a semantic framework, named SMF (semantic matching framework), which brings out the handicap situation for each user in a given environment and allow him to identify the accessible services in this environment. We chose OWL-DL to represent the contextual information which is adequate for knowledge sharing, machine processing and reasoning

3. Semantic Matching Framework (SMF)

The core functionality of SMF is based on a semantic matching between a user model and an environment model. The user model characterizes users' factors, for instance the user's name, his preferences and capacities; those factors are defined as user's attributes. The environment model describes environment factors it specifies devices (e.g., doors, windows, sensors, etc.), defined as "effectors", each effector contains a set of characteristics defined as environment's attributes, for instance, the effector *door* includes the attributes: "required force to open the door", the "door size", etc. We studied this factors (user and environment) and extract some of them who, after association (what we name "*semantic matching*"), can leads the user to the handicap situation. This matching is formalized under a set of first logic rules involving attributes from both models (user and environment).

SMF aims to deliver assisted services according to the user's capabilities using a reasoning process. It Analyses the user and the environment attributes, then infer all handicap situations in the environment that helps to deliver personalized services to the user. The reasoning uses the set of first order logic rules that involves the user and environment characteristics defined above.

4. Data representation

4.1 User and environment models

In order to provide a uniform way to make the different entities understanding each other and to describe the semantics of the attributes (which we name as Meta-Data), we used ontology to represent the domain knowledge (DO). Using ontologies, we described the semantic of both models (user and environment models). Elements of user and environment models are designed by classes, subclasses, attributes and services (see UML diagrams of figure 1) a user model contains classes or a set of attributes, on the other side, environment model contains classes, subclasses and effectors, each effector is represented by a set of attributes and services. Figure 2 gives an example of the data representation of SMF using ontology; classes represented by ellipses; user attributes and effectors are defined by rectangles. An instance of the user or the environment model is called a **profile**. It is implemented as an object which is an instance of a class where the attributes are instantiated by values. For instance in figure 2, the class "tap cold water" is a subclass of the classes "bathtub" and "wash basin". Those classes are subclasses of the class root "Environment". The class "taps cold water" contains two attributes "Required hand force" and "Required hand workspace" and contains also the services Open/Close.

The use of ontologies in this context requires a well-designed and compatible ontology language supporting well-defined semantics and powerful reasoning tools. The syntax of this language must be both intuitive to human users and compatible with existing standards (such as XML, RDF, and RDFS). Its semantics should be formally specified since it could not otherwise provide a shared understanding. Finally, its expressive power should be adequate i.e. the language should be expressive enough for defining all the relevant concepts in enough detail. OWL-DL language [14], which is based on description logic, should be an ideal candidate. It can provide high quality ontologies for our domain knowledge and allows us to formalize and implement the matching on first order rules and reason on it. In this paper we will focus on the implementation using OWL-DL of the domain knowledge of SMF. We invite readers to consult [15] to see in details our approach to define the formal description of the different entities of our domain knowledge (DO) using Description Logic. OWL-DL is an ontology language for the Semantic Web technology. It is developed by the World Wide Web Consortium (W3C) Web Ontology Working Group [14].

The user and the environment models are defined as OWL-DL classes. Table 1 presents the corresponding representation on OWL-DL of the different entities of our DO.

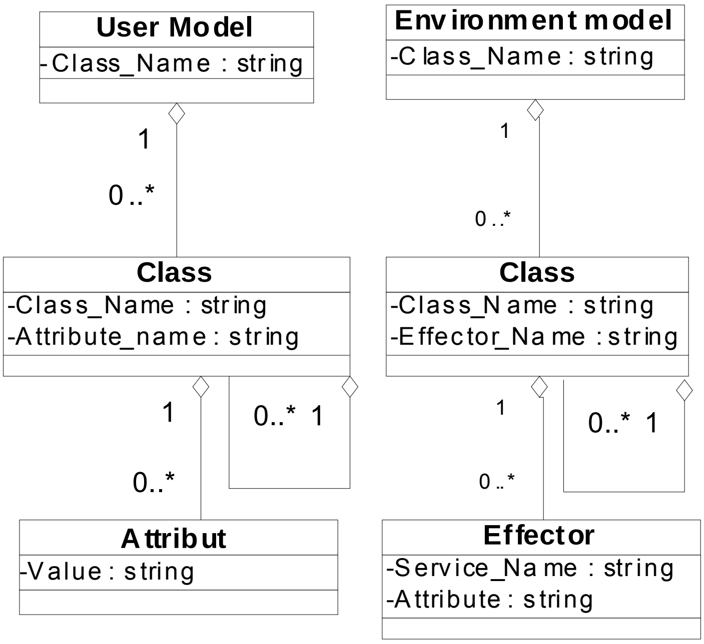


Figure 1. UML Diagram of User and environment Models

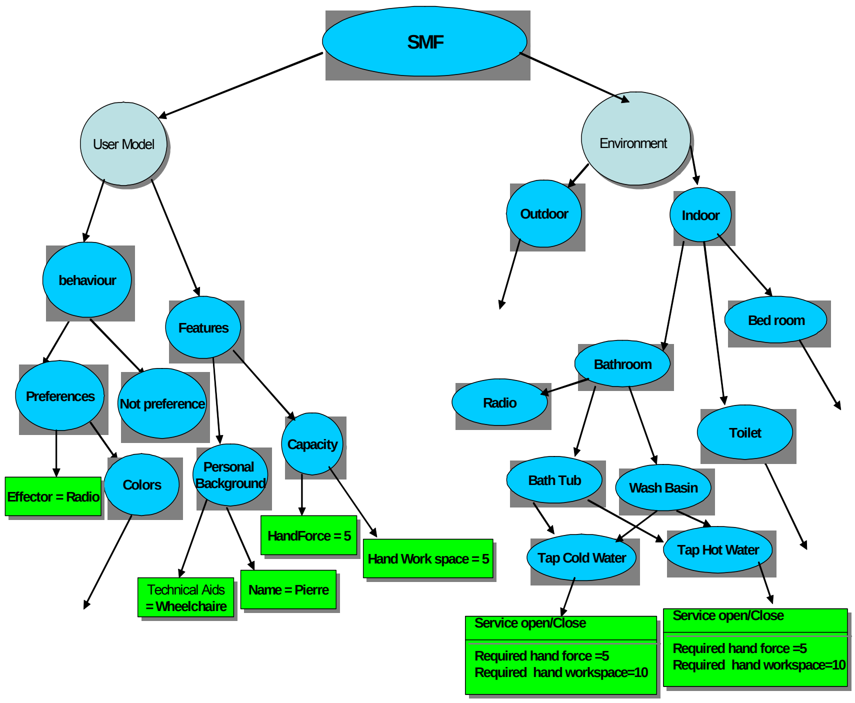


Figure 2. Example of SMF ontology

Table1. SMF ontology on OWL-DL

Ontology element	OWL-DL representation
User model	<owl:Class rdf:ID="*Users*"/>
Environment model	<owl:Class rdf:ID="*Environnement*"/>
Attributes	<owl:DatatypeProperty> Or <owl:ObjectProperty>
User Profile	<Users rdf:ID="*Pierre*">
Environment Profile	<Environnement rdf:ID="*Salon*">

4.2 Semantic matching

The semantic matching between attributes defines all cases that lead the user to the handicap situation. It is represented by a set of first logical rules called "*matching rules* ".
Here we have an example of a set of rules that identify a handicap situation in a given environment.

1. *HandicapSituation* (*x*," Stairs") ← *Users*(*x*), *TypeOfTechnicalAids*(*x*,*a*) , *equal*(*a*,"Wheelchair")
2. *HandicapSituation* (*x*," Lamp") ← *Users*(*x*), *TypeOfDisability*(*x*,*a*), *equal*(*a*,"Blind")
3. *HandicapSituation* (*x*," Radio") ← *Users*(*x*), *TypeOfDisability*(*x*,*a*), *equal*(*a*,"Deaf")
4. *HandicapSituation* (*x*,y)← *Users*(*x*), *HandForce*(*x*,*a*), *Effector*(*y*), *RequiredHandForce*(*y*,*b*), ¬*less_than*(*b*,*a*)
5. *HandicapSituation* (*x*,y) ← *Users*(*x*),*HandWorkspace*(*x*,*a*), *Effecteur*(*y*), *RequiredHandWorkspace*(*y*,*b*), ¬*less_than*(*b*,*a*)

The predicate *less_than* defines the Boolean function *lessThan(x,y)* which returns true if $x \geq y$ and false else. The predicate *equal* defines the Boolean function *equal(x,y)*, it returns true if $x = y$ and false if not.
The rule number 1 defines the matching between the effector "Stairs" of the environment and the attribute technical aid "Wheelchair" of the user *x*. It means, when we have a context where the environment contains stairs and the user is using a wheelchair, this context leads the user to a handicap situation.
To complete our Data representation and have an automatic reasoning over DO, we need to define a property named *HandicapSituation* which should be related to the user class and contains the set of effectors that leads the user to a handicap situation.

4.3 The property *Handicap_situation*:

It is a user attribute; semantically it defines effectors that lead the user to a handicap situation. It is defined by the following owl:ObjectProperty

```
<owl:ObjectProperty rdf:ID="Handicap_situation">
    <rdfs:domain rdf:resource="#Users"/>
    <rdfs:range rdf:resource="#Environment"/>
```

```
</owl:ObjectProperty>
```

The implementation of SMF integrates a reasoning module which processes iteratively the semantic matching rules and the OWL-DL classes' instances (user and environment) and updates the values of the *HandicapSituation* property by adding the new effectors that lead the user to handicap situation.

5. Semantic Matching Framework architecture

The SMF framework is composed of several collaborating modules (Fig. 3): knowledge base ontology (KBO), an instantiation manager, inference reasoner, and a query engine. They operate with external data (user profile and context manager) and provide tasks and services.

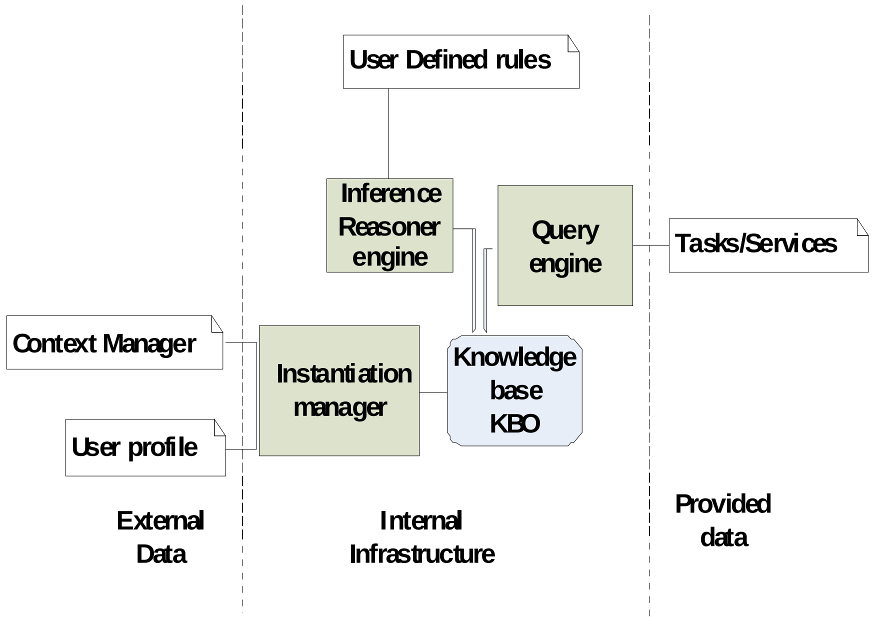


Figure 3. SMF architecture

Once the user is detected as well as his location, the client application in the Framework exchanges information with the instantiation manager module in order to update the KBO which contains the user and environment instances (called profiles).
The Inference Reasoner Engine module, using the program rules, iteratively processes the KBO instances and matching rules to perform the matching between the users and effectors that lead him to the handicap situations. The Query Engine module, using declarative queries, extracts services that do not lead the user to handicap situation from the updated KBO.

5.1 SMF Class Diagram

We implemented the framework using Java2 platform. The class diagram in figure 4 shows in details the cohabitation and implementation of SMF modules.

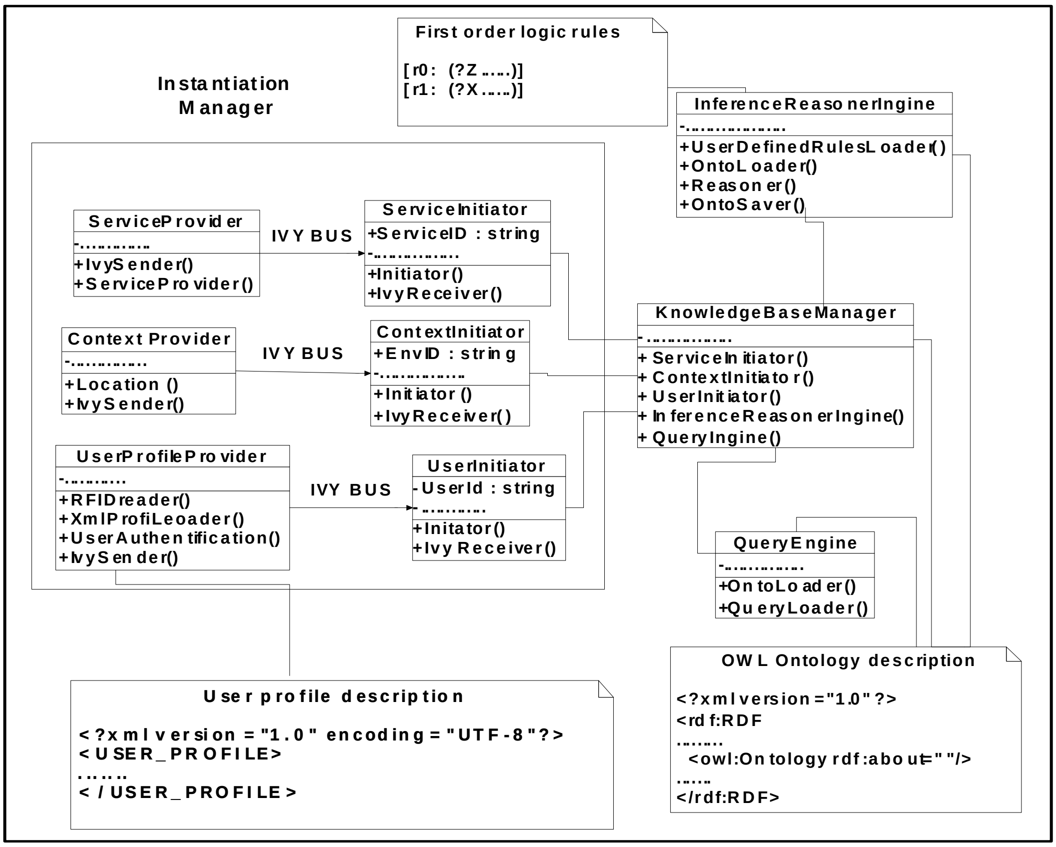


Figure 4. SMF Class diagram

The different modules are developed as classes, the knowledgeBaseManager() class is the heart of the framework. It invokes the different modules on three successive phases:

1. The initiation manager does the initiation phase. It receives external data through the Ivy event-based bus [16]. It is composed on three classes:

- The servicesInitiator() class initiates the environment services; received from the ServicesProvider() class which provides events regarding the environment's effectors and their services and characteristics.
- The ContextInitiator() class updates the contextual information which are received from the ContextProvider() class which integrates all information related to the user's context such as, time, user's location, sensors states, etc.
- The UserInitiator() class makes the user identification through the UserProfileProvider() class which contains the user profile.

2. The inference reasoning phase, managed by inferenceReasoerIngine() class, process iteratively the OWL-DL ontology description base and makes the inference reasoning through the semantic matching rules.

3. The extracting of relevant services from the OWL-DL ontology description base, using declarative queries via the queryEngine () class.

SMF is designed to be integrated in general smart home demonstrator operating with other frameworks to ensure a context aware application. SMF receives, by the Instantiation manager, events from these Frameworks. Thus we adopted a standard way to exchanges data between these Frameworks. For instance, the user profile is defined in a file, using XML standard, as follow:

```
<?xml version="1.0" encoding="UTF-8"?>
<USER_PROFILE>
  <CLASS>
    <Features>
      <Personal_Background Name="Pierre"
       Type_of_disability="SCI_C5/6" />
      <Capability Hand_Force="5" hand_workspace="3"/>
      <Technical_Aids Type="Wheelchair" />
    </Features>
    <Bihaviours>
      <Preferences>
        <Effectors_Preferences Prefer="Coffee" />
      </Preferences>
    </Bihaviours>
  </CLASS>
</USER_PROFILE>
```

The UserProfileProvider() class receives the XML file and, throw initiator() function, update or creates a new user instance in KBO. By the same way ServicesProvider() and ContextProvider() classes update the KBO with the new initiated values, coming from other frameworks.

5.2 Reasoning

The reasoning process is leaded by the inference reasoning engine module implemented using the JENA 2 semantic reasoner [17]. It can be summarized by the follow algorithm:

```
Input: (E,R)(t)
Output: É
Let  V, an empty vector
Do:
É = Reasoner ((E,R),(t))
V=(R,E)(t)
t=t+1
Until V=(E,R)(t)
```

The algorithm has as input the environment context at time *t*, which is defined by the matching rules *R* and the environment instances *E*. The environment instances gather the set of users and environment profiles in the environment. The environment context can change by adding or removing rules or by changing the environment instances. Each time the environment context is modified, the algorithm runs the reasoner.

The algorithm extracts the environment context from the ontology as OWL-DL classes (Figure 5). The matching rules are formatted as Jena rules standard [17]. The inference engine is started using the reasoner API of JENA (Φ). The result (output) is checked with the Inputs (point fix checking). If they are different, the reasoner runs again, otherwise the reasoner stops. This result is then used to update the ontology.

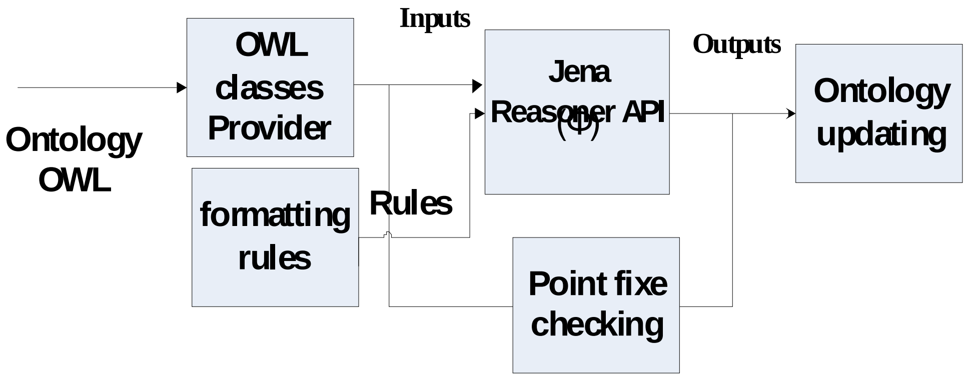


Figure 5. Architecture of the inference engine module

5.3 Multi-User Management

SMF is designed to support multi users; each user is processed separately by a session and delivered by adapted services. First we highlight the different steps regarding one session, and then we define, in details, how SMF modules mange the multi users process.
Steps of one user session are shown in the activity diagram of Figure 6. At the first we have the user authentication step. An environment access decision is taken according to the user profile. If the user is not allowed to access the environment's services, it is then sent back to the final state of the diagram and informed why he was denied access. If he gets permission, two concurrent threads are started: one for services discovery process and the other for user localization. Then at the end of both threads, the system precedes the Inference reasoning step. The final step is service delivery. In the same time and if another user goes inside the environment, the same process is triggered concurrently, and a new user session begins to carry out the various phases of the services delivery process.

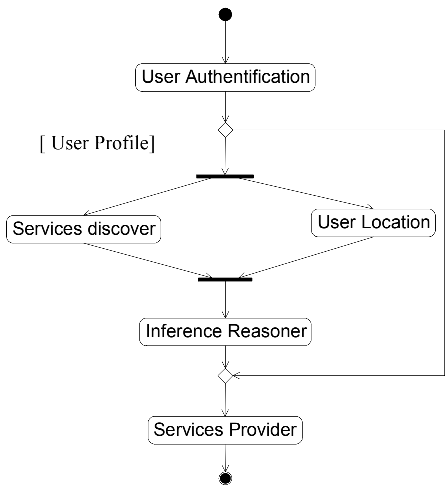


Figure 6. Activity diagram

Figure 7 shows how SMF modules mange the multi-user process. The first step is the user's identification (1). Once that phase is completed, the user will be informed of the outcome (2): If he is allowed to access the environment, his profile will be sent to the Knowledge Base Manager Module (4) Through the Initiator Manager module (3). Then the Knowledge Base Manager creates, for that user session a thread (5), which loads and sets up the profile in the KBO (6). As soon as the user leaves the environment, the User Identification module inform the initiation manager by the state of the user (7) which forwards it to the Knowledge Base Manager (7) in order to stop the thread from that session (9) and updates the KBO (10)

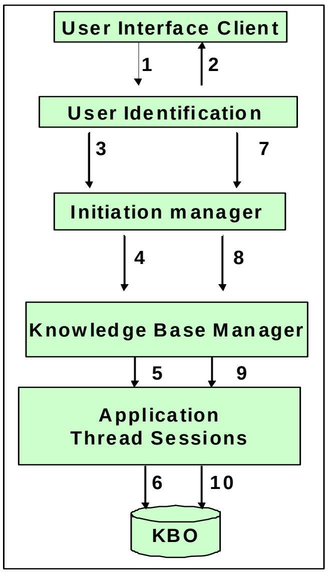


Figure 7. Multi-user management

6. Experimentation

We build a prototype in our test experimentation site (Laboratory conditions). We implemented a context aware application demonstrator for smart environment dedicated to people with disabilities integrating the context aware Framework (CAE) [18], the Service Management Framework (SEMF) [19], and of course SMF. The demonstrator is designed on client-server architecture. Mobile handhelds (PDA, Tablet PC…) are used as clients. Their low processing capacities advocated for a thin client approach. So clients mainly convey inputs and outputs between the user and a remote server.

All SMF modules were implemented on a windows machine. RFID Development kit (reader + software + set of tags) was integrated for user authentication and identification. Infra Red motion detectors sensors were used in order to perform user localization (indoor). Within indoor environment we have integrated Power Lines Communication (PLC) X10 protocol to control homes appliances (Lights, Coffee machine, etc.). Outdoor, there was a Bluetooth access point which ensures network access and process outdoor user localization.

Besides experimenting the SMF framework, the demonstrator ensures the services continuity supported by the SEMF framework which is tested by the user movement trough both environment indoor and outdoor [20].

Users with motor disabilities were provided with a Tablet PC mounted on assistive platform (wheelchair + Manus robot [21]) in order to evaluate and test the feasibility of this prototype (figure 8).

Wifi technology is used to establish wireless connection to the local network. The client is installed in a tablet PC representing the user terminal. Besides a Wifi card, this terminal has also a Bluetooth plug to connect to the outdoor environment simulated by a local Bluetooth network.

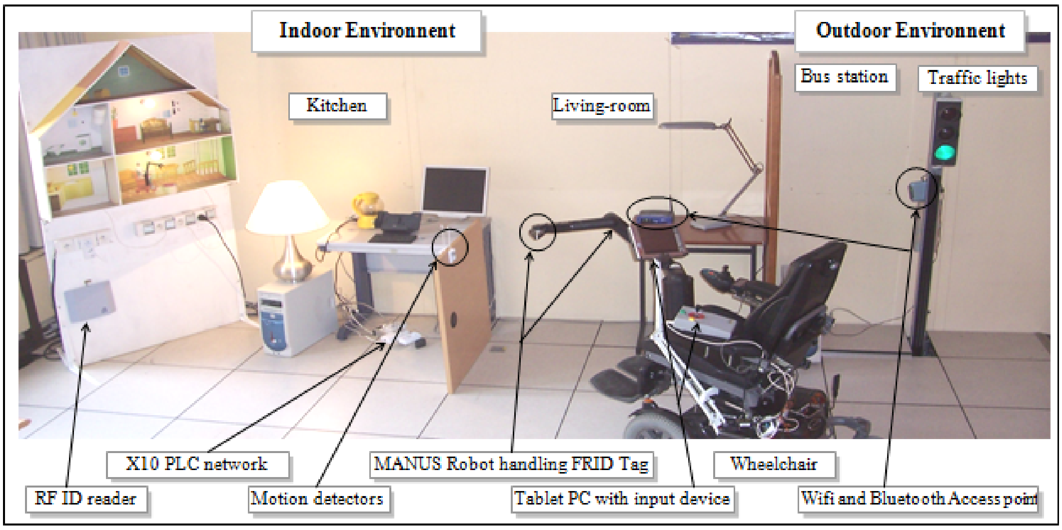


Figure 8. General demonstrator

The context aware Framework (CAE) integrates a sensors model which is dynamically updated with events send from RFID readers, X10-PLC, Infra Red motion detectors, etc. This model sends events via IVY bus to the SMF Instantiation manager module [22].

Regarding the Service Management Framework, it receives the data from SMF which are the services that do not lead the user to handicap situation. Then this Framework, using an OSCAR platform [23] (which is a java-based OSGI [24] technology), delivers the accessible services to the user. The services deployed in this experimentation are defined in the following table:

Table 2. Tested services in the demonstrator

Environments		Services
Home	Kitchen	- Lamp1 - Coffee-machine
	Living-room	- Lamp2 - radio
Road	Road	- Bus schedule - Traffic lights control

Once user is detected and the location of the user is identified, the context aware Framework (CAE) exchanges information with the SMF instantiation manager module in order to instantiate the KBO. The Inference Reasoner Engine module, using the matching rules, iteratively processes the KBO instances and performs the matching between the user and the environment. The Context Query Engine module, using declarative queries, extracts the accessible services from the KBO and provides them to the Service Management Framework (SEMF) which adapts the application in the user terminal side.

7. Results

In order to evaluate the performance of the reasoning response time of SMF regarding supporting multiple users, we varied the number of rules and user instances in the KBO. The system was implemented on a 2.4 GHZ processor with 1.0 Gbyte of RAM running on Windows XP. The response time of the Reasoner is expressed in milliseconds (ms). Graph *a* of figure 9 shows the time response for a set of rules fixed to 10 with the number of users varying from 10 to 60. The graph *b* of figure 9, presents the time response for a fixed number of user instances (fixed to 60) with the number of rules varies from 10 to 60. We remark that, for both graphs (*a* and *b*) varying the number of user instances and the number of rules from 10 to 60 gives a gap of 800 ms (RT increases from 400 ms to 1200 ms) ms for graph *a* and a gap of 200ms (RT increases from 1050 ms to 1250ms) for graph *b*. Consequently, changing the number of rules does not have much impact on the reasoning time as much as increasing the number of user instances. So, in order to improve the response time of SMF, we should decrease the number of user instances in the Knowledge base KBO, for instance eliminating the instances of users who have left the environment.

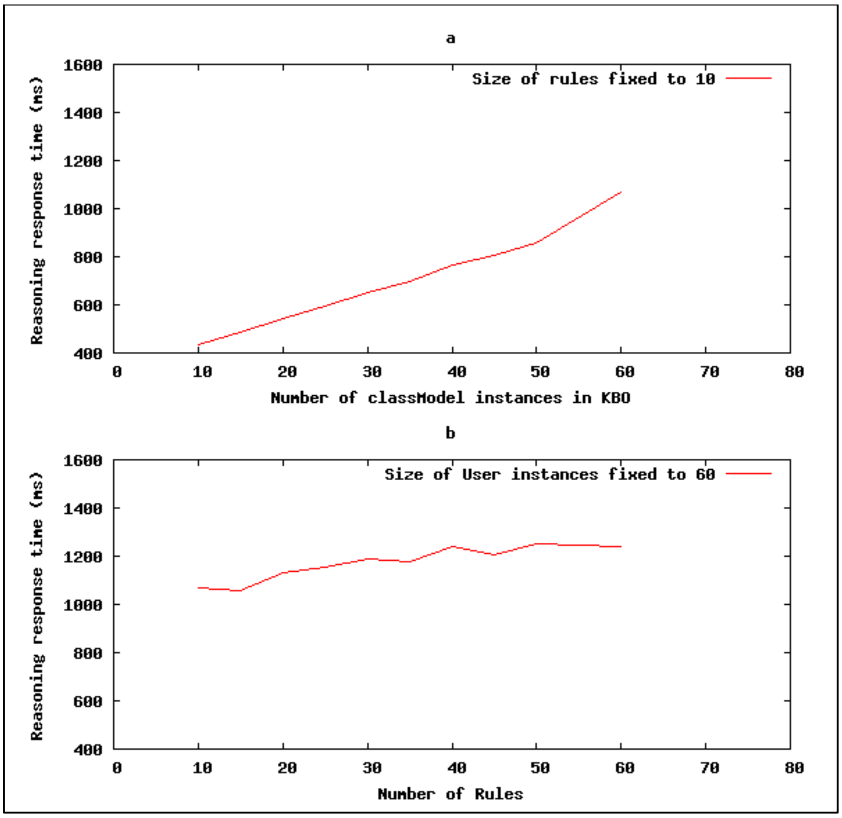


Figure 9. Performances of the Reasoning process

8. Conclusion

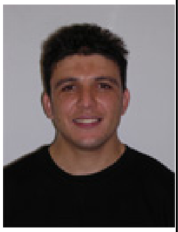
In this paper we have presented a general smart home demonstrator including a semantic matching framework (SMF), to detect, in the environment, the user accessible services. SMF is based on the detection of handicap situation on this environment. Particularly we presented the data representation of SMF using OWL-DL and the definition of matching rules based on first order logical rules. The paper includes our approach for multiple user management where each user is assigned to a session, and the design of a session is based on threads implementation. We have evaluated the performance of the system on multiple users context, the system time response depends closely on the number of user's session. We are aiming at validating this concept in real environment involving end-users by deploying this platform within a pilot site (residence site). In the near future, we are working to integrate another module to handle unknown environment services and enable managing non-deterministic environments, since living environment are multiple and unknown by the system (ex. Friend house, public space, etc.).

References

[1] N_House, " http://architecture.mit.edu/house_n/web,"
[2] C. D. Kidd, R. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. D. Mynatt, T. Starner and W. Newstetter, "The aware home: A living laboratory for ubiquitous computing research," in *CoBuild '99: Proceedings of the Second International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture,* 1999, pp. 191-198.
[3] P. F. e. Al, "Classification québécoise: Processus de production du handicap," RIPPH,SCCIDIH; Réseau international du processus de production du handicap, 2000.
[4] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes and F. Potter, "Place lab: Device positioning using radio beacons in the wild," in *Proceedings of Pervasive,* 2005, pp. 116–133.
[5] G. Ferguson, J. Allen, N. Blaylock, D. Byron, N. Chambers, M. Dzikovska, L. Galescu, X. Shen, R. Swier and M. Swift, "The medication advisor project: Preliminary report," 2002.
[6] MARC, "http://marc.med.virginia.edu/,"
[7] C. D. Kidd, R. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. Mynatt, T. E. Starner and W. Newstetter, "The Aware Home: A Living Laboratory for Ubiquitous Computing Research," *LECTURE NOTES IN COMPUTER SCIENCE,* pp. 191-198, 1999.
[8] G. D. Abowd, E. D. Mynatt and T. Rodden, "The Human Experience," *IEEE PERVASIVE COMPUTING,* pp. 48-57, 2002.
[9] G. D. Abowd, A. Bobick, I. Essa, E. Mynatt and W. Rogers, "The aware home: Developing technologies for successful aging," in *AAAI Workshop and Automation as a Care Giver,* 2002,
[10] H. Pigot, D. Lussier-Desrochers, J. Bauchet, Y. Lachapelle and S. Giroux, "A smart home to assist recipes' completion." in *Festival of International Conferences on Caregiving, Disability, Aging and Technology (FICCDAT), 2nd International Conference on Technology and Aging (ICTA),* June 16-19, 20072007, Toronto, Canada
[11] H. Kautz, L. Arnstein, G. Borriello, O. Etzioni and D. Fox, "An overview of the assisted cognition project," in *AAAI-2002 Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder Care,* 2002, pp. 60–65.
[12] E. Dishman. (2004, Inventing wellness systems for aging in place. *Computer 37(5),* pp. 34-41. Available: http://domus.usherbrooke.ca/articles/dishman04.pdf
[13] D. B. Gray and G. E. Hendershot, "The ICIDH-2: Developments for a new era of outcomes research," *Arch. Phys. Med. Rehabil.,* vol. 81, pp. 10-14, 2000.
[14] World Wide Web Consortium (W3C), "OWL Web Ontology Language Guide," February, 10. 2004.
[15] R Kadouche, M Mokhtari, S Giroux, B Abdulrazak , "Semantic Approach for modelling an assistive environment using Description Logic" The 10th ACM, International Conference on Information Integration and Web-based Applications & Services (iiWAS2008), Austria.
[16] S. Chatty, "The Ivy software bus-a white paper," *CENA Technical Note NT02-816, from Http://www.Tls.Cena.fr/products/ivv,* 2002.
[17] Anonymous "Jena - A Semantic Web Framework for Java," vol. 2007,
[18] M. A. Feki and M. Mokhtari, "Context awareness for pervasive assistive environment," in *Handbook of Research on Mobile Multimedia* ,Idea Group Publisher ed.Anonymous Ismail Khalil Ibrahim, 2006,

[19] M. Ghorbel, M. Mokhtari and S. Renouard, "A distributed approach for assistive service provision in pervasive environment," in *Proceedings of the 4th International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots WMASH '06,* 2006, Los Angeles, USApp. 95.
[20] M. Ghorbel, R. Kadouche and M. Mokhtari, "User & service modeling in assistive environment to enhance accessibility of dependent people," *ICTA, April,* 2007.
[21] M. Mokhtari, B. Abdulrazak, R. Rodriguez and B. Grandjean, "Implementation of a path planner to improve the usability of a robot dedicated to severely disabled people," *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on,* vol. 3, pp. 4023-4028 vol.3, 2003.
[22] M. A. Feki, R. Kadouche and M. Mokhtari, "Integration of sensor's context into assistive environment for people with severe disabilities," in 2005,
[23] R. S. Hall and H. Cervantes, "An OSGi implementation and experience report," in *IEEE Consumer Communications and Networking Conference,* 2004, pp. 5.
[24] OSGi Alliance, "OSGi Service Platform Core Specification - Release 4,", 2005.

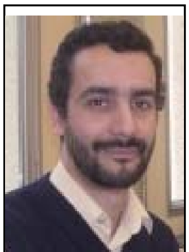
Authors



Rachid Kadouche, PHD, is holding a post-doctoral research fellowship at the University of Sherbrooke, Canada. He received his Ph.D. in computer science from Telecom-SudParis and university of Evry in 2007.In 2003 he obtained his Master degree from University Denis Diderot Paris7, France.
His research activity focuses on User modelling design in case of dependent peoples and development for pervasive software applications. He is also interested in human machine interaction.

Bessam Abdulrazak, PHD, is an Assistant Professor of Computer and Science and Engineering at the University of Sherbrooke, Canada. Previously, he was a Post-Doc at the university of Florida (US), and before that at Telecom-SudParis (France). He received his Ph.D. in computer science from Telecom-SudParis and university of Evry, M.S. of Robotics from Paris VI university (France), and B.Sc. Ing, from USTHB (Algeria). His research interests include ubiquitous and pervasive computing, ambient Intelligence, smart spaces, assistive technologies, rehabilitation robotics and health telematics.

Sylvain Giroux, PHD, is a professor at the University of Sherbrooke, Canada. He received a Ph.D. in computer science from the University of Montréal, in 1993. He worked in close collaboration with the Center for Advanced Studies, Research and Development in Sardinia (CRS4), Italy. He co-founded the DOMUS laboratory of the University of Sherbrooke in 2005. His main research interests are mobile computing, pervasive computing, distributed artificial intelligence, multi-agent systems, user modelling, intelligent tutoring systems.



Mounir Mokhtari, PHD, is an associate professor at National Institute of Telecommunication (INT) in France. He obtained his master in physical science from university de val de marne France in 1992, then a master in networking from INT (1994). He collaborates with university Pierre and Marie Curie to obtain his PHD in computer Science in 1997. Mounir Mokhtari's research activity focuses on human machine interaction, rehabilitation robotic and health thematic. He obtained his Qualification of Research Director from the University Pierre and Marie Curie, in Computer Sciences, November 2002.