Service matching in online community for mutual assisted living

Service Matching in Online Community for Mutual Assisted Living

Hong Sun, Vincenzo De Florio, Ning Gui, Chris Blondia
Department of Mathematics and Computer Science
Performance Analysis of Telecommunication Systems group
Middelheimlaan 1, 2020 Antwerp, Belgium, and
Interdisciplinary institute for BroadBand Technology
Gaston Crommenlaan 8, 9050 Ghent-Ledeberg, Belgium
{hong.sun, vincenzo.deflorio, ning.gui, chris.blondia}@ua.ac.be

Abstract

Elder people are becoming a predominant aspect of our societies. As such, solutions both efficacious and cost-effective need to be sought. Our previous study suggests that mutual assisted living community, where dwellers make contributions to the community. Thus community resources could be fully utilized and requested services would be provided in a prompt way. We used service oriented architecture (SOA) to orchestrate the available resources inside the community; and we organize the services into different forms such as formal ones (carried out by professionals, etc), informal ones (assisted by relatives, volunteers, .etc) and even as group activities. We expect such a community could help not only efficiently utilizing the social resources in maintaining the independent living of the elderly people, but also helping these people maintain their connections to the society and bring them entertainment, so that the quality of their living standard may be improved at the same time. This paper continues the track of the previous studies on building mutual assistance communities. Service matching algorithms are developed providing flexible choices on both requested services content and the organization forms of the services.

1. Introduction

As well known, the proportion of elderly people keeps increasing since the end of last century. The European overview report of Ambient Assisted Living (AAL) investigated this trend [1]. The studies of EUROSTAT [2] indicated that the share of the total European population (EU 15) older than 65 is set to increase from 16.3% in 2000 to 22% by 2025 and 27.5% by 2050, while that over 80 (3.6% in 2000) is expected to reach 6% by 2025 and 10% by 2050.

Studies of Counsel and Care in UK found out that elderly people would prefer to live in their own house rather than in hospitals, thus they need support to remain independent at their home [2]. Furthermore, researches also found that remote clinical therapy at home does not bring negative effect to the therapy process [3]. In order to improve the quality of life for the elderly and disabled people, it is important to guarantee that assistance to those people be timely arranged in case of need.

Assistive devices are developed to facilitate the daily lives of these elderly and disabled people. But assistive devices also have their limitations: For instance, in the AAL country report of Finland, it was remarked that “the (assistive) devices are not useful if not combined with services and formal or informal support and help” [4].

We share this view and deem that human resource is still indispensable. We have developed a design tool to evaluate performance of informal carers in so-called mutual assistance communities [5], i.e. communities whose members may request assistance and at the same time get motivated to play the role of caregiver. Simulations have shown that informal carers are indeed capable to contribute effectively to the community welfare. We have also introduced the “participant” model to encourage the elder people to participate in group activities they are able to, and change their role from passively requesting to actively participating [6]. The simulations also demonstrate that the participant model can help to effectively utilize the social resources and increase social connections.

As our preliminary investigations illustrate, coordinating the requests of the elder people in a

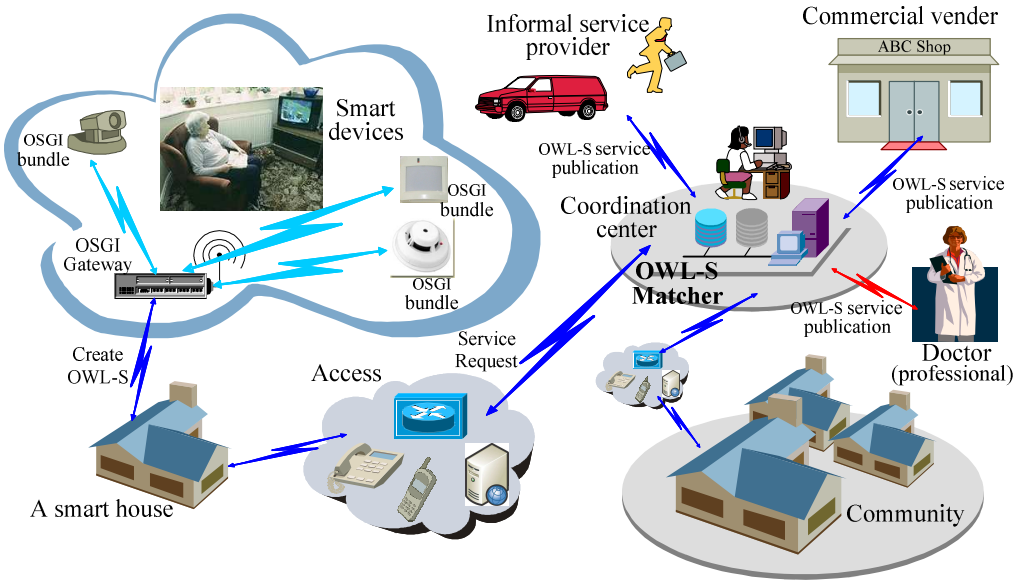

Figure 1. Organization of Mutual Assistance Community

community view is the best way to utilize the social resources. Hence we have developed an infrastructure using service oriented architecture to build up ambient assisted living community by processing the requests in formatted services [7], however, the details of the service matching algorithm is not implemented inside that paper. In this paper, we will introduce the method to reason about the advertised and requested services.

We are going to use the OWL-S [12] to specify the advertised and requested services. The service matching is based on mapping the types of the services, and mapping the organizations of services. The type of services focus on the functionalities of the services, while the organizations of services focus on how to organize the services – this means in which form the services are provided. By specifying the service organization, the mutual assistance community could be organized with the presence of different service forms, such as help from informal carer, participant to group activities, etc. Based on the hierarchical architecture, which is a natural characteristic of the ontology, we use the OWL-S Matcher [14] for mapping the advertised and requested service, and providing the degree of the mappings.

The rest of the paper is organized as follows: in Section 2, we introduce the community structure for ambient assisted living. The matching algorithms are introduced in Section 3. Experiments and results are shown in Section 4. Future works and a scenario are presented in section 5, and conclusions are given in Section 6.

2. Mutual assistance community

As mentioned in the introduction, our previous research shows that constructing mutual assisted living communities to provide health service to the elderly people may be an effective approach to save considerably the social resources. This conclusion was reached by extensive simulations of the effects of such a community. In this section, we introduce the design of such a mutual assistance community with the infrastructure of service oriented architecture.

The structure of the proposed community is shown in Fig. 1. Assistive devices will be deployed to construct a smart house environment around the assisted people. These assistive devices will be developed as OSGI bundles and be managed by a local OSGI gateway [17]. Based on the information from the assisting devices, the local coordinator could send alarm signals if the assisted people are in a dangerous situation.

The most important asset integrated in this community, we think, is the people themselves. The community allows disparate technologies and people working together to helping people who suffer from aging or disabilities. People who are able to provide services are encouraged to do so and assist the requesting people as informal carers. They are encouraged either to help their relatives who required help or to help the other needed people as volunteers. Elder people are also encouraged to participate in the group activities, which not only helps to maintain physical and psychological health but also reduces the requests of professional medical resources. Professional carers (such as doctors, specialists etc.) are included in the community to provide emergency and professional medical service. Commercial vendors are also included in, which brings convenience to the user and diversifies the service

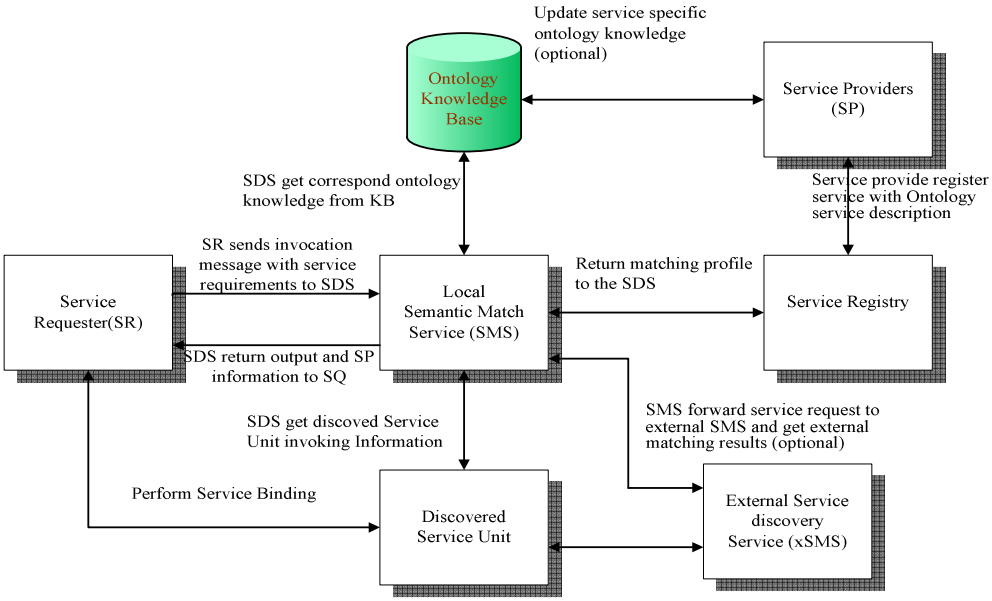

Figure 2. Software Architecture

type, at the same time laying the foundation for economical exploitation. The processes of sending requests, receiving help, or organizing group activities are executed by the software infrastructure in the service oriented approach. Based on our previous analysis, we designed a software architecture, which is shown in Fig. 2, aiming to provide infrastructural support for the AAL community. The architecture is mainly consisting of three parts: service registration, service matching and service binding.

The most difficult part is the service matching, which maps the requested and provided services. As the service provider is a very important concern in our mutual assistance community, the mapping process should not only base on the functions of the services, but should also include organizing the services in different forms, such as informal service, participating group activities, etc. As the description of the web service might be vague and unclear, the mapping algorithm should also take this uncertainty into account. In the following session, we will introduce the mapping algorithm that meets the above mentioned challenges.

3. Matching Web Services

Service publication and service matching are the most important issues in organizing our proposed mutual assistance community. Researches on web service discovery and mapping are becoming hot research topics which aim to processing the increasing web services over the Internet. This section will introduce the related techniques for service description and service matching, especially the OWL-S Matcher, which will be employed in our architecture for service matching.

3.1 Semantic Services

Industry efforts to standardize web service description, discovery and invocation have come to standards such as WSDL [8] and UDDI [9]. However, these standards, in their current form, suffer from the lack of semantic representation. The notion of Semantic Web services [10] [11] takes us one step closer to interoperability of autonomously developed and deployed Web services, where a software agent or application can dynamically find and bind services without having a priori hard-wired knowledge about how to discover and invoke them.

OWL-S [12] is a specific OWL [13] ontology designed to provide a framework for semantically describing Web services from several perspectives, for instance, service inquiry, invocation, composition. The information necessary for Web service discovery would be specified at the service sites as computer-interpretable semantic markup, and a service registry or ontology-enhanced search engine could be used to locate the services automatically.

The existing OWL-S matching algorithms are mostly based on comparing the information contained in the OWL-S profile, mainly by the inputs and outputs of a service. In our mutual assistance community, the matching process is focused on comparing the functionality of the service and the organization forms of the service. In order to meet the challenges brought by the heterogeneity, and uncertainty of the ontology description, we also

expect the matching algorithm could reflect the degree of the matching result.

3.2 OWL-S Matcher

OWL-S Matcher [14] is a JAVA implementation of a matching algorithm for matching requested and advertised OWL-S services by comparing service descriptions. The mapping process is executed by running the embedded JessKB reasoner to compare the items in the requested and advertised services. Based on the ontology hierarchy of the compared entries, the results will indicate the relationship of the compared item requested and advertised service.

In this paper, we will use the OWL-S matcher to implement the service match process. In the service matching process of our proposed mutual assistance community, the comparison will focus on the Service Type (indicating the function of a service) and Service Provider (indicating the service organization).

Matching Degree

Matching degree is an essential part in the OWL-S Matcher [15]; it reflects the relationship of the ontology in the requested and advertised services. By providing the matching degree of the assigned ontology, the relationship between the requested and advertised services could be obtained. Let A and B denote two concepts of ontology in the requested and advertised services. The matching degree, in other words, the relationship between A and B could be defined as in Table 1:

Table 1. Matching Degree

Rank	Degree of match	Explnation
0	Fail	The requested and advertised class do not match.
1	Unclassified	At least one of the classes is unclassified.
2	Type_invert	The class in the advertised service is the sub-class of the one in the requested service.
3	Type_subsumes	The class in the requested service is the sub-class of the one in the advertised service.
4	Match	The requested and advertised class match.

4. Experiments and results

In the experiments, some primitive hierarchical ontology is constructed to represent different service types and different service providers. Web services are also constructed representing requested and advertised services respectively. Mapping process

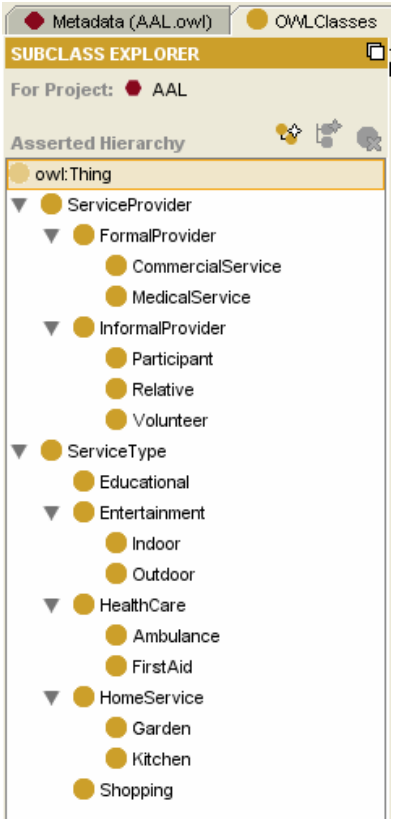

Figure 3. Hierarchical Classes in Protégé

```
</profile:textDescription>
<profile:hasInput  rdf:resource="#_InformalProvider"/>
<profile:hasOutput  rdf:resource="#_Entertainment"/>

<profile:has_subclass rdf:resource="AAL_SERVICE_PROCESS" />

<profile:has_process rdf:resource="AAL_SERVICE_PROCESS" />
</profile:Profile>

<process:ProcessModel rdf:ID="AAL_SERVICE_PROCESS_MODEL">
<service:describes rdf:resource="#AAL_SERVICE_SERVICE"/>
<process:hasProcess rdf:resource="#AAL_SERVICE_PROCESS"/>
</process:ProcessModel>

<process:AtomicProcess rdf:ID="AAL_SERVICE_PROCESS">
<process:hasInput  rdf:resource="#_InformalProvider"/>
<process:hasOutput rdf:resource="#_Entertainment"/>
</process:AtomicProcess>

<process:ServiceType rdf:ID="_Entertainment">
<process:parameterType rdf:resource=
"http://127.0.0.1/ontology/my_ontology.owl#Entertainment" />
  <rdfs:label></rdfs:label>
</process:ServiceType>

<process:ServiceProvider  rdf:ID="_InformalProvider">
<process:parameterType rdf:resource=
"http://127.0.0.1/ontology/my_ontology.owl#InformalProvider" />
  <rdfs:label></rdfs:label>
</process:ServiceProvider>
```

Figure 4. Fragment of the Advertised Service in OWL

between the requested and advertised services is executed to test how the different degrees of mapping could be obtained).

Figure 3. illustrates some primitive hierarchical concepts that were built in Protégé [16]. The concepts are made up by two main classes: Service Provider (formal provider, informal provider, etc.) and Service Type (educational, healthcare, etc.).

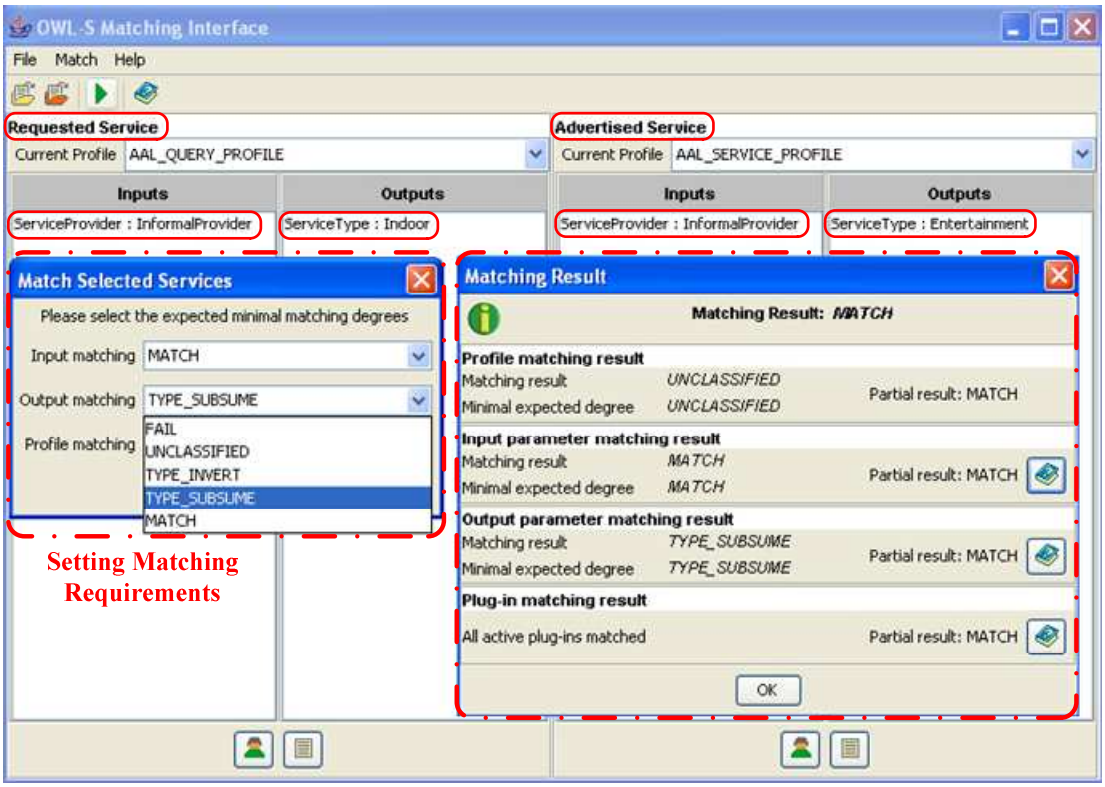

Figure 5. OWL-S Matcher Interface and Matching Result

After the definition of the hierarchical classes, the advertised and requested services are written in OWL using the classes in Fig. 3. Both services are very simple and only specify the Service Provider and Service Type.

In the requested service, the requester is asking the informal provider to provide some indoor entertainment service. In the advertised service, the service provider is indicated as informal provider, and the available service is indicated as entertainment service.

Figure 4. shows a fragment of the advertised service in OWL. The #Entertainment ontology and the #Informal Provider ontology are declared in the my_ontology.owl file and specified as inputs and outputs of the service. This is because the OWL-S Matcher could assign different matching acceptance for inputs and outputs, and separating the service provider and service type enables the service requester to specify different acceptable matching degrees for service type and service provider separately.

Figure 5. shows the interface of the OWL-S Matcher. Before running the matching process, the advertised and requested services should be loaded respectively. When the matching process is executed, a window (in the central left part of Fig. 5) will pop up and ask the service requester to set the requirements for the matching result. The matching degrees shown in this window are explained in Table 1. In the settings shown in Fig. 5, the user selects

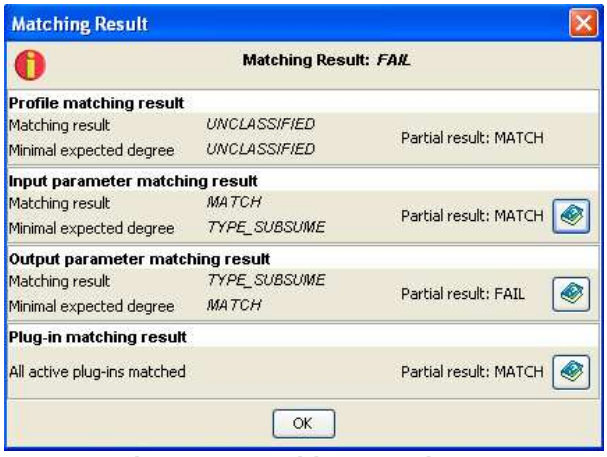

Figure 6. Matching Result II

'Match' as acceptance degree for the input (ServiceProvider) and 'Type_subsumes' for the output (ServiceType). The profile matching is not used in this experiment, so it is assigned as "Unclassified". After setting the acceptances and clicking the button "run", the matching process is then to be executed. The matching result is shown in the central right window in Fig. 5. It indicates that for the input (ServiceProvider), the matching result is "Match". That is because in both advertised and requested service, the input is set to Informal Provider. The result "Match" meets the minimal expected degree, which is also "Match" in this case. So that the partial result is "Match", which indicates

the expected matching degree is met. For the output (ServiceType), the matching result is "Type_subsumes", as the requested one "Indoor" is a subclass of "Entertainment" (see in Fig. 3); and the partial result is also "Match".

Figure 6. shows the matching results after the selection of the minimal expected degrees. It can be seen that the input matching result "Match" still meets the expect degree "Type_subsumes"; while for the output, the result "Type_subsumes" can not meet the expected degree "Match", so that the final result for the output matching is "Fail".

The experiments show that through the OWL-S Matcher, matching process between the advertised and requested services could be effectively executed. Relationships between the requested service and those advertised could be achieved in the matching process. Service requesters could set different expected matching degrees for the required service. By separating the service function (Service Type) and service organization (Service Provider), the requester could specify different acceptance degrees on function and organization, which increases the flexibility in the matching process. The service organization mode developed in our previous researches, such as organizing community with the attendance of informal carer [5], as well as the introduction of group activities where the elder people could join as peer participants [6], could also be realized by choosing appropriate service provider.

5. Future Work and Application Scenario

The future work will focus on bringing such a mutual assistance community into real application. Firstly, we will enrich the ontology for describing the environment of the mutual assistance community. With the detailed ontology definition, more parameters could be included in the service description, such as specifications of location, deadline of the services, etc. These specified requirements can bring out more accurate matching process.

Secondly, the automatic invocation mechanism of web services should be developed, so that the assistive devices could be manipulated remotely. These assistive devices would be wrapped as different services in order to implement the service oriented architecture (SOA) framework shown in Fig. 2 by orchestrating these services and bringing the blueprint in Fig. 1 into real-life.

Thirdly, we deem that the user friendliness is crucial when applying this system to the elder people. Currently, the service publication and service request are achieved by loading the owl file which is written in Fig. 4. Writing the owl file is extremely cumbersome for the users, and is absolutely not appropriate for the elder people. Graphical user interface will be developed to provide easy access to make a request or advertisement. We expect the above mentioned developments could construct a mutual assistance community for assisting the elder people in their daily life. We envisaged some scenarios that the elder people get their required services; we will present one of such scenarios in the following:

Scenario:

Mary is 70 years old and lives alone in Antwerp. In the afternoon of a sunny day, Mary wants to have a walk in the Middelheim Park, which is close to her home. She wants to have somebody to accompany with her during the walk. Mary decides to use the mutual assistance community to find someone who also wants to have a walk in the park. She switches on the TV, which is the graphical interface of the mutual assistance community. She navigates the service menu, which is built as an ontology tree, and selects the "Group Activity". A few photos will be presented to her, representing group activities such as chatting, exercise together, etc. Mary chooses the symbol of walking; she types in the location she wants to hold this event as "Middelheim Park". For the service organization, Mary chooses to receive service from peer "participants"; she specifies the deadline as "today, 8pm", and she chooses to forward the service requests to "relatives" in case that no such service is found till the deadline.

After making such inputs, the service matching engine will start searching for the appropriate services. The service matching engine is located in the service matching center. Available services are advertised in the service matching center. The service matching engine will produce a list of available services, and rank them by their relevance to the users' requirements. This way the system can best fit the users' requirements. The settings for the search, translated from Mary are as follows:

Required Service: Walk with someone.
Location: Middelheim Park, Antwerp.
Deadline: 8pm, 1st, August, 2007.
Service Form: Participant of Group Activities.
Action if no match found: Forward to "relatives"

After a few seconds, the screen shows the searching result. "*Mrs. Brown* initiates a *walking* activity in *Middelheim Park* by *7pm today*; if you agree to take this *group activity* with *Mrs. Brown*, press the **confirm** button and we will forward *Mrs. Brown* your contact information."

Mary agrees to join this activity, she pressed "confirm", while at the other side, Mrs. Brown also confirms the participation of Mary. Their contact information is then displayed on their TV screens

respectively, they call each other to confirm the time and place to meet, and have a nice time walking in the park later.

The above mentioned scenario is one of the applications that could be carried out by our mutual assistance community. We expect such a community could help the elder people actively participate in group activities, to maintain their social connection, build their self-esteem, fight against the feeling of lonely and save the social resources in providing care to the elder people.

6. Conclusions

The constant increase of the population of elder people poses enormous economic and social challenges to our society. This paper proposes to build up an online mutual assistance community, where the contributions from the dwellers could greatly alleviate the burden on social security system. Group activities for the elder would be highly encouraged so that their social connections are kept and where they are respected as peer participant.

Service publication and matching are big challenges for building such an online community. In this paper, we tested the feasibility to use the OWL-S Matcher for service publication and matching. Experiments show that through such an approach, matching results could highlight the relationship between the requested and advertised service by indicating a matching degree. By separating the service function and service organization, the requester may set different acceptance degrees for service function and service organization respectively. By strengthening the mapping in service organization, help from informal carers, or participants for a group activity could be easily described and found. And we expect such activities could save the medical resources in assisting the elder people, help to maintain social connections for the elder people and bring them comfort and well being.

Future work will focus on building detailed ontology library for the environment of the mutual assistance community; develop the user friendly graphical interface; implement remote control of the assistive devices with OSGI framework, and implement the service oriented software framework to bring the proposed system into real application.

7. References

[1] H. Steg, et al. Ambient Assisted Living – European overview report, September, 2005.

[2] Counsel and Care, Community Care Assessment and Services, April, 2005.

[3] J. Deutsch, J. Lewis, and G. Burdea, Technical and patient Performance with a Virtual Reality-Integrated Telerehabilitation System, IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 15(1), pp. 30-35, March 2007.

[4] Ambient Assisted Living, country report, Finland, 2005.

[5] H. Sun, V. De Florio and C. Blondia. A design tool to reason about Ambient Assisted Living Systems. In the Proceedings of the International Conference on Intelligent Systems Design and Applications, Jinan, China, 2006.

[6] H. Sun, V. De Florio, N. Gui and C. Blondia. Participant: A New Concept for Optimally Assisting the Elder People. In the Proceedings of the 20th IEEE International Symposium on Computer-Based Medical Systems (CBMS-2007), Maribor, Slovenia, 2007.

[7] N. Gui, H. Sun, V. De Florio and C. Blondia. A Service-oriented Infrastructure Approach for Mutual Assistance Communities. Proceedings of the First IEEE WoWMoM Workshop on Adaptive and DependAble Mission- and bUsiness-critical mobile Systems (ADAMUS 2007), Helsinki, Finland, 2007.

[8] E. Christenson, F. Curbera, G. Meredith et al. Web Services Description Language (WSDL), http://www.w3.org/TR/wsdl.

[9] UDDI Technical Committee., Universal Description, Discovery and Integration (UDDI), 2000.

[10] T. Berners-Lee, J. Hendler, and O.Lassila, The Semantic Web, *Scientific American*, May 2001.

[11] L. Li and I. Horrocks, A Software Framework For Matchmaking Based on Semantic Web Technology, 12th Intl. Conference on World Wide Web, 2003.

[12] OWL-S Technical Committee (T.C), Web Ontology Language for Web Services (OWL-S), *http://www.w3.org/Submission/OWL-S/*, 2002

[13] M. Dean, G. Schreiber, S. Bechhofer, et al. OWL Web Ontology Language Reference, Feb.2004

[14] Stefan Tang, Christoph Liebetruth. The TUB OWL-S Matcher. http://owlsm.projects.semwebcentral.org.

[15] Michael C. Jaeger and Stefan Tang, Ranked Matching for Service Descriptions using DAML-S. CAiSE'04 Whorkshops, Riga, Latvia, June, 2004

[16] M. Horridge, H. Knublauch, A. Rector, R. Stevens and C. Wroe. A Practical Guide To Building OWL Ontologies Using The Prot´eg´e-OWL Plugin and CO-ODE Tools Edition 1.0, University of Manchester, Stanford University, 2004.

[17] Open Service Gateway initiative (OSGI), OSGI Service Platform Version 4, http://www.osgi.org, August, 2007.