



A UML Extension for Modeling Web Services Communities

Hela Limam^{*}, Mouna Gharsalli, Jalel Akaichi

Department of Computer Science
High Institute of Management, France

Abstract— *The growth of Web services in the information space raises difficulties. These services are developed by different entities and there is no consensus on the way to use services descriptions. Web services are created, updated or deleted on the fly. Moreover, those changes may be not reported in services registries for these reasons, Web services may present many performance problems. Publish, discover and compose services in such an environment raises a number of problems. Services research must be done in a large unstructured and changing set of Web services. Hence, managing this set of Web services will considerably ameliorate the process of Web services discovery and selection. In this context, the concept of community appears a solution for structuring and a organizing a large set of unstructured Web services sharing the same area of interest. Indeed, the community infrastructure contributes to improving the availability of Web services and provides centralized access to distributed Web services. However, the organization into communities raises many problems ranging from community development to communities management: how to specify and how to manage Web services that reside in a community. Although Web services are extensively studied problems communities have not been fully addressed. Hence, we propose in this paper a UML profile for specifying and modeling structural, behavioral and interactional aspects of Web services communities.*

Keywords— *Web services community, UML profile, model, management.*

I. INTRODUCTION

In collaborative environment, Web services seem to be a privileged means for interconnecting applications in organizations. As the number of Web services published on the Internet continues to grow, the need for organization and integration of these services becomes a necessity. Hence the emergence of the concept of Web services community that aims to bring together a potentially large number of Web services according to their common features in order to facilitate and improve the process of Web service discovery in a large environment such as the Internet. Indeed, the infrastructure of community helps to improve the availability of Web services and provides a centralized Web services through distributed access. However, the organization community raises issues ranging from community development to communities management operations. Indeed, one of the main issues raised by the communities of Web services is the design problem resulting from the lack of a generic tool for modeling Web services communities. In this work we propose a generic design tool for modeling communities of Web services namely a UML (Unified Modeling Language) profile for modeling Web services communities. This profile consists of the extension of use case diagram, class diagram and sequence diagram modeling the structural, behavioral and interactional aspects of Web services communities. This paper is organized as follows. In section 2 we introduce the concept of Web services community and we present an overview of Web services community management approaches and we provide a comparison of these approaches. In section 3 we present our Web services communities profile stereotypes. Behavioral, structural and interactional model of Web services communities are described in section 4, 5 and 6. In section 7 we illustrate our approach by a case study related to a health care community. Finally, in section 8 we conclude the work, and give the direction of our future works.

II. RELATED WORKS

In addition to the competition that lies between Web services, collaboration between them has become increasingly necessary as the needs of users are complex beings and a single Web service can not satisfy. Hence the interest of the combination of Web services known as community structures [1]. Indeed, communities of Web services enable the integration of several Web services on the same field, allowing new opportunities for interaction and provide a breeding ground for interoperability. The main objective of a community of Web services is to provide a framework for research and dynamic selection of Web services [3]. It is also a way to support dynamic composition and allow the substitution of services [4]. In general, the term community refers to "a group of people who live together and / or are linked by common interests, religion, nationality, etc." [5]. When it comes to Web services, a community of Web services can be composed of Web services offering the same functionality or sharing similar concerns. Members of a community can compete with each other when offering similar services or can complement each other when offering additional services. A community is a collection of Web services that have the same functionality even though they have no distinct functional properties [6]. In [7], a community is seen as a way to offer ontology of Web services with the same area of interest. The authors in [8] use the community to implement techniques based on the rules for comparison of contextual policies of Web services. The authors in [1], a community is seen as a way to provide a common description of a desired

functionality without explicit reference to a specific Web service. Wan et al. [9] define communities of Web services as virtual spaces that can dynamically gather a different web services having complementary or related functionality. The authors in [10], use communities to group records of Web services. Indeed, a community of Web services registries is a set of records with Web services that provide similar functionality. The concept of Web services community highlights the importance of management and organization of Web services in a community. Indeed, it raises several problems such as how to specify and implement a community, how to manage Web services that reside in a community and how to resolve conflicts within a community and between communities. These issues were addressed in different ways in different approaches.

Several approaches focused on Web services communities' management. Benatallah et al [3] [11] propose the Self-Serv platform that consists of a share in a service composition environment (also called the service manager) for the definition and deployment of composite services and communities of secondly in a runtime environment that acts as middleware for orchestrating composite services and perform dynamic sourcing. The platform also offers graphical tools to (1) identify the pattern of composite services; (2) the deployment of these specifications on a network, (3) the execution of composite services deployed instances, and (4) management communities of Web services. Self-Serv three types of services: basic services, composite services, and community services. A basic service is a point of access to an application that does not rely on another web service to meet the demands of users. A composite service is a structure that includes other elementary and composite services that work together to implement a set of operations. The concept of community service is a solution to the problem of the composition of a potentially large number of dynamic Web services. In [12] WS-catalogNet is a framework for organizing and integrating a large number of catalogs that are combined in a container, also known as community associated with a specific domain. To make available their products, vendors must register their catalogs with a community. WS-catalogNet is a web service that allows the creation of communities, joining a catalog with a community and defining relationships between these catalogs. The goal is to provide a model to query (online) community through relationships between the different catalogs of the community. This template allows you to set up web portals for end users. The authors used an architecture based on the pair-to-pair approach. Medjahed et al [13] proposed the WebBIS system as a generic framework for defining and managing the composition of Web services in dynamic environments. In this framework, Web services are semantically organized in terms of pull-push-communities and communities. Both communities establish static and dynamic relationships between Web services. A WebBIS-SDL is proposed to describe, compose, and monitor Web services. The authors in [7] propose an architecture-based community for Semantic Web services. They introduced the concept of community to address an ontological organization of Web service. In fact, they have developed the ontology community that serves as a model for describing Web services communities. For community management of Web services, they proposed an approach pair to pair. This approach has two agents: agent and community members who are responsible for managing changes in the dynamic environment of agent communities.

Maamar et al [14] [15] proposed an approach that supports the concepts, architecture, operation and deployment of Web services communities. They proposed a centralized community by identifying two types of Web services architecture, namely master and slave. The community is accessible via a single master Web service. This work details the management tasks including a master Web service is responsible. The master Web service supports multiple responsibilities: it attracts Web services for part of the community, it means the slave Web services involved in the provision of services on user requests, monitors performance and takes action failure. Also in [14], they designed the protocol Web Services Community Development Protocol (WSCDProtocol) for the management of communities in terms of how to attract, retain Web services. The main purpose of this protocol is to oversee the operations leading to the establishment of a community and managing its content.

The authors in [18], [19] presented a framework for the specification and management of communities of Web services. This framework shows how software agents, able to argue, negotiate, and reason about Web services can be used to specify these Web services and to manage their respective communities. The use of argumentative agents making independent Web services and allows them to better manage their roles in a community being better organized within communities and in achieving the objectives for which they are designed. Each Web service is associated with an argumentative agent. The community is led by a master Web service, which, among others attracts new Web services to the community, retains existing Web services in the community, and identifies the Web services in the community that will participate in the compositions of Web services. All these operations are managed through the interaction of agents through flexible conversations made by argumentation, persuasion, and negotiation phases called dialogue games.

Benharref et al [20] proposed a managerial community of Web services for managing communities of Web services. This community is able to monitor, certify and assess the quality of Web services from other communities. A Web service will be allowed in the community if the quality of service is high or rejected if the quality of service is low. In this paper, the authors focused on the use of the managerial community in the selection of web services based on quality of service. In [10] Bouchaala et al proposed a graph-based approach for managing communities of Web services registries. This approach consists of a set of algorithms and operations management. Management operations are verified at each stage of the life cycle of the community (create, delete, merge, split). The algorithms are rather defined steps for managing the life cycle of records (join the network, join a community, update functionality). These algorithms and these operations are tested and validated using the simulation graphs.

We compare the approaches mentioned above according to the following criteria, as shown in Table 1:

- Concepts: concepts considered in each approach cited
- Architecture: the architecture proposed by the model approach

- Collaboration: the possibility of interactions between web services within the community.
- Semantic: the representation of the semantic Web services and the community.
- Composition: the combination of features of several Web services.

TABLE I
COMPARISON OF MANAGEMENT WEB SERVICES COMMUNITIES' APPROACHES

Approaches	Concepts	Architecture	Semantic	Collaboration	Composition
[3], [11]	Community of services	Peer to Peer	No	Yes	Yes
[7]	Ontology community	Peer to Peer	Yes	Yes	Yes
[10]	Graphs based approach	Master- slaves	Yes	No	No
[12]	Community of catalogs	Peer to Peer	Yes	Yes	Yes
[13]	Pull-community Push-community	Peer to Peer	Yes	Yes	Yes
[14], [15]	WSCD Protocol	Master- slaves	No	No	Yes
[16]	Coopetition intra-community Competition inter-community Reputation	Master- slaves	No	No	Yes
[18], [19]	Argumentative Agents	Master- slaves	Yes	Yes	Yes
[20]	Managerial Community Quality of services	Peer to Peer	No	Yes	No

Examining the work cited above, we notice that they focus on managing Web services communities. They differ in terms of their perspectives and concepts considered (Table 1). Indeed, the proposed approaches are generic and their exploitation is limited to conditions defined in their implementation. Therefore, it is necessary to propose a general approach for modeling and managing Web services communities. Our contribution is to propose a UML profile community Web services which models the behaviour, the structure and the interactions of Web services communities.

III. WEB SERVICES COMMUNITIES PROFILE STEREOTYPES

To provide a formal description of the Web services community, we have adopted the Model-Driven Architecture (MDA) [22]. In the MDA-based approaches [23], the focus is on software development through the development of Unified Modeling Language (UML) models. UML is a graphical language for specifying, visualizing, constructing, and documenting the artifacts of software systems. It is a modeling language standard approved by the OMG [24] and plays a central role in the MDA. Despite the fact that UML is mainly used with object-oriented systems, it has the ability to develop other types of systems through its extension mechanism or profiles. Indeed, UML can also be used as a meta-modeling language, where UML diagrams are used to formalize the abstract syntax of another modeling language. Therefore, we propose a UML profile based on the extension of UML use cases, the UML class diagram and UML sequence. In the following diagram (Fig. 1), we summarize the profile in a package that shows the items that have been extended. The stereotype « community » is used to represent a community that includes Web services related to the same domain.

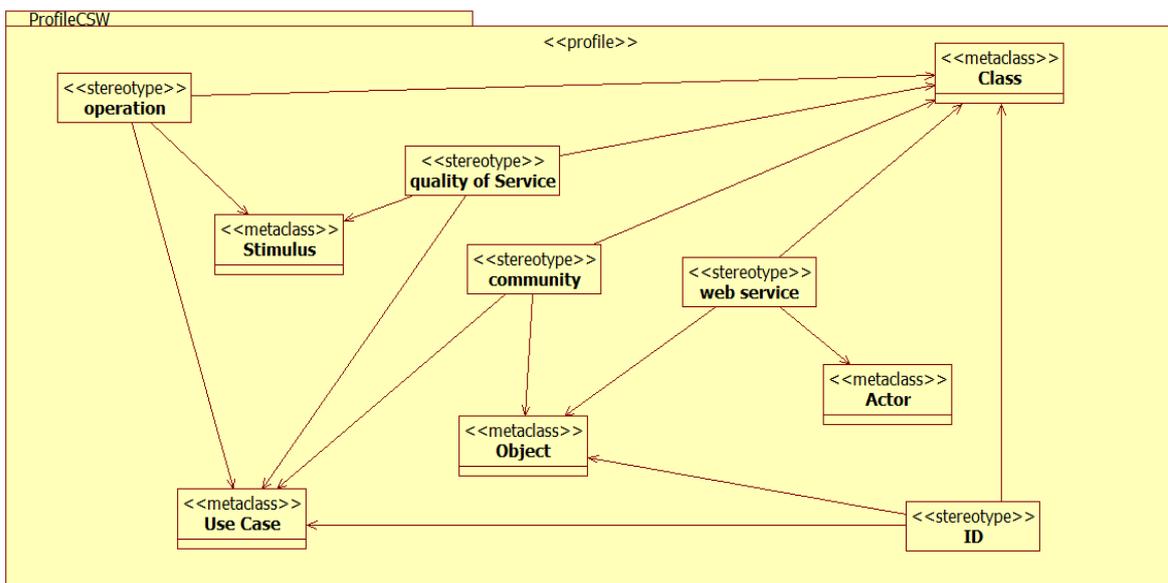


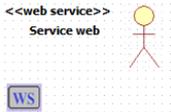
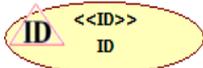
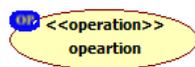
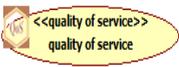
Fig. 1 The stereotypes used in the profile

A community is identified by the stereotype « ID » which represents the identification data of the community such as its name and its class. The stereotype « Web service » is an abstraction of Web service providers, who after recording their Web services become members of the community. A community is accessible via a set of operations. The stereotype of « operation » represents the abstract operations that summarize the main features of the community. The stereotype of « quality of service » specifies a set of quality requirements on the behavior of a community.

IV. THE BEHAVIORAL MODEL OF WEB SERVICES COMMUNITIES

We extended the use case diagram to represent the behavioral aspects of a Web service community. We defined four stereotypes as meta-class « Use Case » and a stereotype as having meta-class « Actor » , they are linked by associations. In Table 2 we present these stereotypes.

TABLE II
STEREOTYPES USED IN THE BEHAVIORAL MODEL OF WEB SERVICE COMMUNITIES

Stereotypes	Icons	Descriptions
« web service »		This stereotype is attached to an actor and says he is a Web service.
« community »		This stereotype is attached to a use case and indicates that this is a case of using community.
« ID »		This stereotype is attached to a use case and indicates that this is a case identifier community use.
« operation »		This stereotype is attached to a use case and indicates that this is a use case operation of a community.
« quality of service »		This stereotype is attached to a use case and indicates that this is a case of using quality of service of a community.

To illustrate the behavioral model created using the profile, we modeled the Web services community. The diagram 'CSWUseCaseDiagram management CSW' in Fig. 2 describes the main requirements for community management of Web services that are defined in the following:

- Creating Web services community: the community manager creates a community by grouping related Web services in the same domain, then it sets the pattern for a description of the field which the community belongs without reference to the Web service providers.
- Establishment of relations between communities: communities can create relationships between them. Relations between the communities are divided into two types: peer relationships, or relationships of specialization.
- Update Web services community: communities operate in a dynamic environment, therefore, communities must be constantly updated. Updating communities takes two forms: the modification and deletion of community.
- Deleting web community services: the community that does not contain a member is removed. Therefore, the community manager will monitor all events happening in the community, for example, it must identify if users constantly leave without performing any action.
- Modification of Web services community: it consists in adding, modifying or deleting a member and therefore the addition, modification or deletion of Web services.
- Registration Web service providers: In order to be accessible through a community, the Web service provider must apply for registration. Registration of Web service is the manager of membership. When the registration request arrives, the manager of associate members of the Web service provider with the relevant community. By registering the Web service provider becomes a member of the community.
- Conversion of Web information sources into Web services: if the source of web information does not take the form of web service, the membership manager must convert to a Web service.
- Definition of community members: The Web service provider defines community members in saving communities, web services related to their areas.

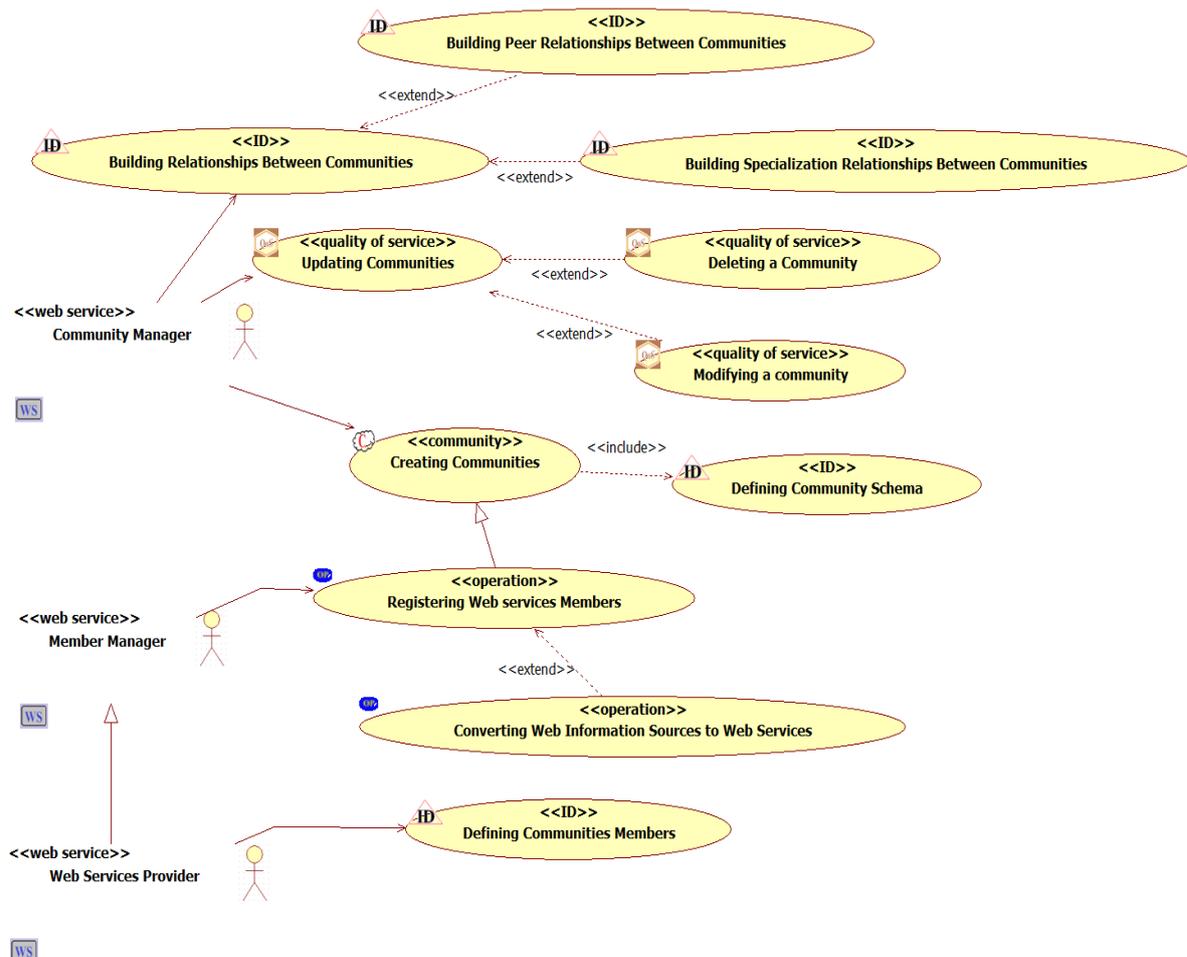


Fig. 2 CSWUseCaseDiagram management CSW

V. THE STRUCTURAL MODEL OF WEB SERVICES COMMUNITIES

We have extended the class diagram to represent the structure of a community of Web service. We defined five stereotypes that are as meta-class « Class » and are linked by associations. In Table 3 we present these stereotypes.

TABLE III

STEREOTYPES USED IN THE STRUCTURAL MODEL OF WEB SERVICES COMMUNITIES

Stereotypes	icons	Descriptions
« community »		This stereotype is attached to a class and indicates that this is a community class.
« web service »		This stereotype is attached to a class and says he is a member of a class community
« ID »		This stereotype is attached to a class and says he is a class identifier of the community
« operation »		This stereotype is attached to a class and says he is a class operation of a community.
« quality of service »		This stereotype is attached to a class and says he is a quality class of service of a community

A community is associated with a specific domain and identified by a class and an identifier. It dynamically collects different Web services with similar functionality. The Web service providers, who register their Web services in a community, are members of the community. A community is accessible via a set of generic operations, community service providers define these operations based on their expertise in the field of community. A generic operation is defined by a set of functional and non-functional attributes. Functional attributes describe the syntactic functions, which represent the structure of a generic operation, and semantic features, which refer to the significance of the operation or its

messages. The semantic attributes are static or dynamic. The quality of operation is based on a set of qualitative attributes which are transverse to the set of operations such as cost and the response time. The proposed structure of Web service community is inspired by the structure of communities of Web service presented in [7]. In Fig. 3 we model the meta-community model of Web services to illustrate the structural model produced using our profile.

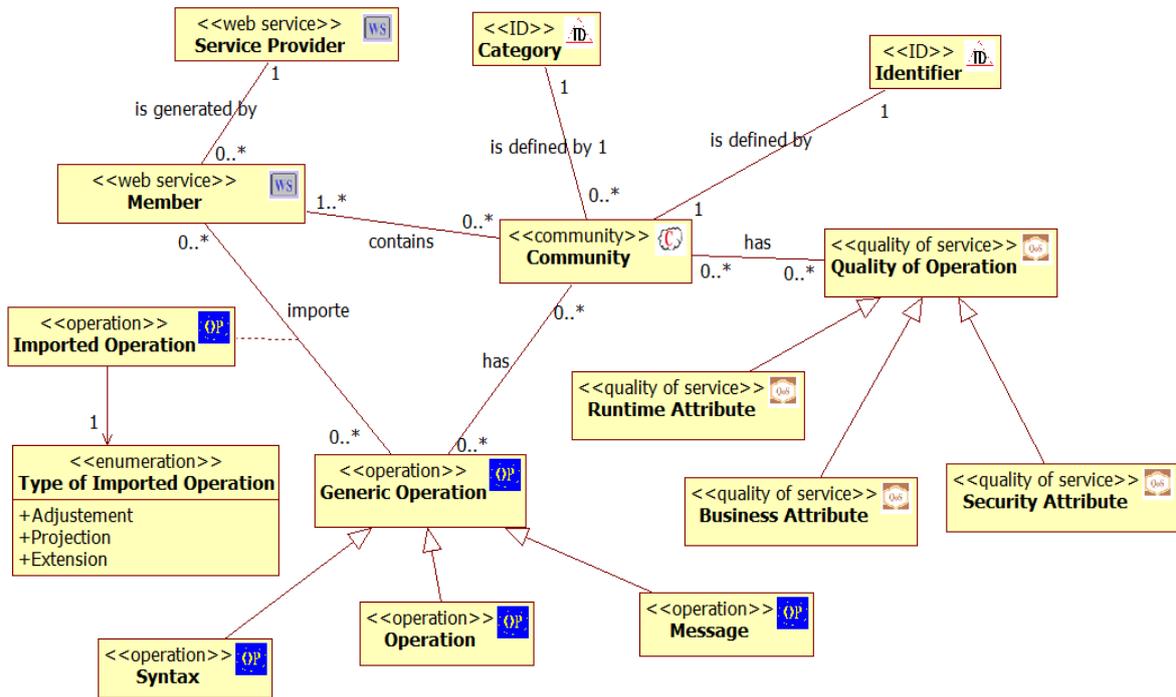


Fig. 3 Meta-model of Web services communities

VI. THE INTERACTIONAL MODEL OF WEB SERVICES COMMUNITIES

We extended the sequence diagram to represent the interactional aspect of a Web service community. This digraph is used to represent the interaction between the elements of a community and between communities. We have defined three stereotypes as meta-class "Object" and communicate with each other through messages with stereotyped as meta-class "Stimulus". In Table 4 we present these stereotypes.

TABLE IV
STEREOTYPES USED IN THE INTERACTIONAL MODEL WEB SERVICES COMMUNITIES

Stereotypes	icons	Descriptions
« community »		This stereotype is attached to an object and indicates that it is a community object.
« web service »		This stereotype is attached to an object and indicates that it is a member of a community object.
« ID »		This stereotype is attached to an object and indicates that it is an object identifier of the community.
« operation »		This stereotype is attached to a message and indicates that it is an operation of a community.
« quality of service »		This stereotype is attached to a message and indicates that it is a quality of service of a community.

To illustrate our interactional model created using our profile, we take the example of modeling querying Web services community. The diagram 'CSWSequenceDiagram query CSW' in Fig. 4 describes the process of user queries is defined in the following:

- Formulate a query: Users can view and access to general information about the community, but they are not able to navigate the system functionality. Their primary role is to request a service.

- Query processing community is made using the following steps:
 - Identify affected communities: Identifying the combination of members with their processing capabilities of the application, when put together, meet all the constraints expressed in the query. Members can be internal (from the community), or external (belonging to peer communities).
 - Rewrite the query to the communities: The application is divided into sub-queries corresponding to the identified communities.
 - Routing requests between communities: The sub-queries are sent to the identified communities.
 - Identify community members affected by the query.
 - Rewrite the query to the members concerned.
 - Finally, provide the results to the user.

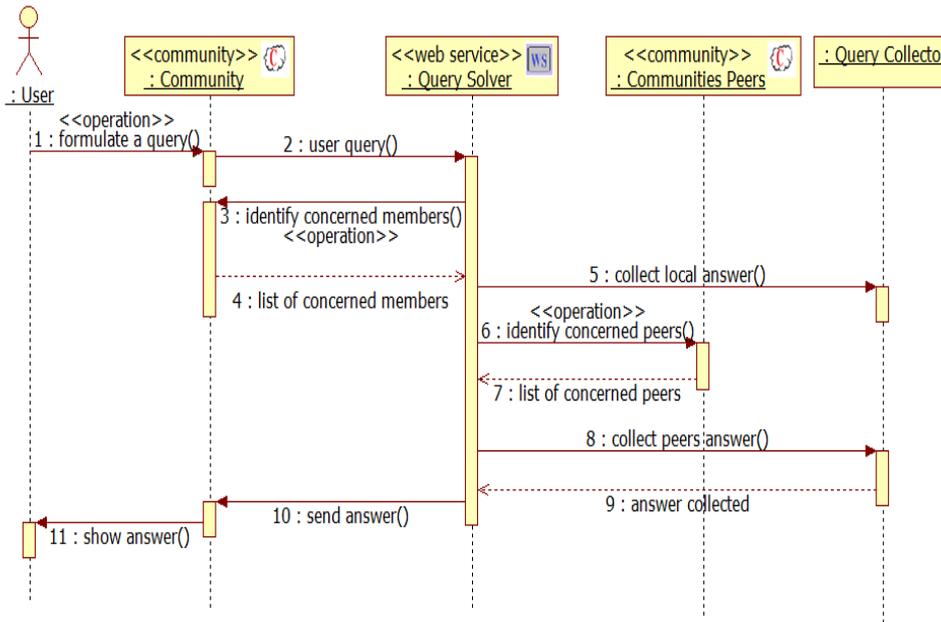


Fig. 4 CSWSequenceDiagram query CSW

VII. CASE STUDY

To implement our approach we chose the open source platform StarUML. In fact, StarUML [25] is a modeling platform software that supports UML (Unified Modeling Language). It actively supports the MDA (Model Driven Architecture) approach by supporting the UML profile. StarUML adds new functions that are adapted to our area of interest.

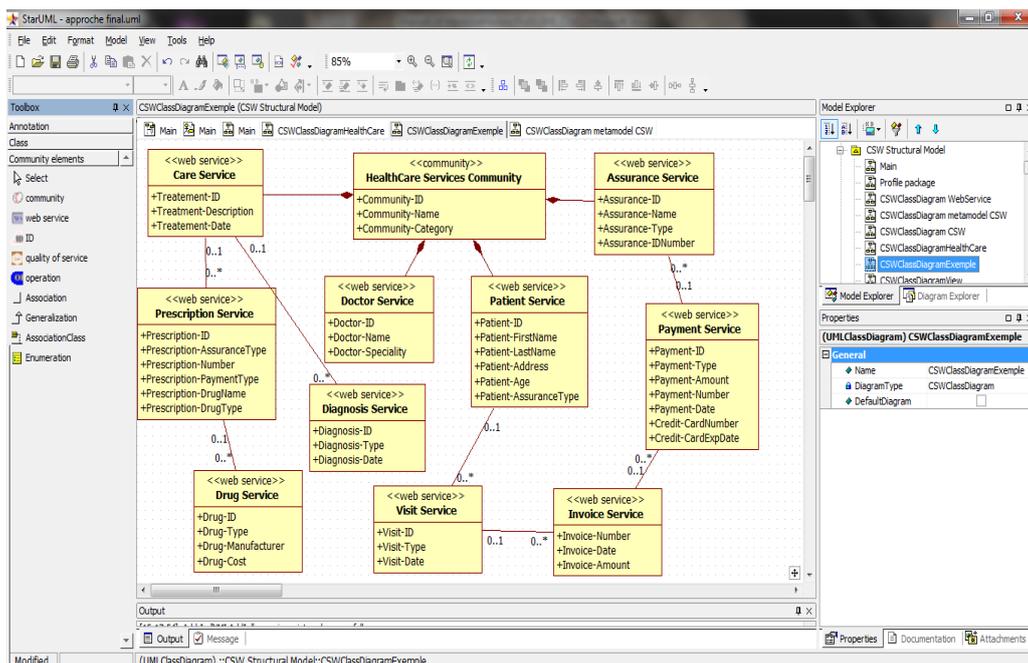


Fig. 5. Detailed CSWClassDiagram HealthCareCommunity

In StarUML platform we added an approach called 'CSW Approach' which is composed of three models: 'Behavioral model', 'structural model' and 'interactional model'. We also created a UML profile community Web service consists of three diagrams which are respectively 'CSWUeCaseDiagram', 'CSWClassDiagram' and 'CSWSequenceDiagram'. To illustrate how our approach, described above, can be applied to different areas we have chosen the example of a community health care. A community health care is created to consolidate services on the field of health. It consists of a set of Web services of health care such as care services, patient care service physicians, prescriptions services, diagnostic services, medication services, insurance services, etc. Web services within a community can be related to each other through relationships, as shown in Fig. 5, which shows the class diagram of community health care.

VIII. CONCLUSION

In this paper, we addressed the problem of designing communities of web services ranging from modeling its specification, its management until its querying. So our goal is to provide a solution modeling community of Web services namely a UML profile. The proposed profile is based on an extension of UML to introduce the concept of community of Web services. Extensions that we have made have been applied to UML use case diagrams, class and sequence to model the structural and behavioral interactional aspect of Web services communities respectively. As prospects, we expect first to move forward with our project including code generation for the deployment of Web services communities.

REFERENCES

- [1] Z. Maamar., M. Lahkim, D. Benslimane, P. Thiran and S. Subramanian, « Web Services Communities - Concepts & Operations – » Proceedings of The 3rd International Conference on Web Information Systems and Technologies (WEBIST), 2007.
- [2] B. Benatallah, Q.-Z. Sheng and M. Dumas, « The Self-Serv Environment for Web Services Composition », IEEE Internet Computing, 7(1), 2003.
- [3] B. Benatallah, M. Dumas and Q.-Z. Sheng, « Facilitating the Rapid Development and Scalable Orchestration of Composite Web Services », Distributed and Parallel Database, Springer Science, pp. 5-37, 2005.
- [4] M.-C. Fauvet, H. Duarte, M. Dumas and B. Benatallah, « Handling Transactional Properties in Web Service Composition », Proceedings of the 6th International Conference on Web Information Systems Engineering 3806, pp. 273-289, 2005.
- [5] <http://www.merriam-webster.com/dictionary/community>, consulted the 10/04/2012.
- [6] M.-A. Serhani and A. Benharref, « Enforcing Quality of Service within Web Services Communities », Journal of Software, 2011.
- [7] B. Medjahed and A. Bouguettaya, « A dynamic foundational architecture for semantic web services », Distributed and Parallel Databases, Vol. 17, No. 2, pp. 179-206, 2005.
- [8] B. Medjahed and Y. Atif, « Context-based Matching for Web Service Composition », Distributed and Parallel Databases, Springer, 21(1), 2007.
- [9] W. Wan, J. Bentahar and A. Ben Hamza, « Modeling and Verifying Agent-based Communities of Web Services », Proceedings of the 23rd International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA-AIE), 2010.
- [10] O. Bouchaala, M. Sellami, W. Gaaloul., S. Tata, M. Jmaiel « Modeling and Managing Communities of Web Service Registries », WEBIST, pp. 88-102, 2011.
- [11] Q.-Z. Sheng, B. Benatallah, M. Dumas and E.-O.-Y. Mak, « SELF-SERV: a platform for rapid composition of web services in a peer-to-peer environment », Proceedings of the 28th VLDB Conference, 2002.
- [12] H. Paik, B. Benatallah and F. Toumani, « WS-CatalogNet: Building Peer-to-Peer e-Catalog », In Proc. of 6th International Conference on Flexible Query Answering Systems, 2004.
- [13] B. Medjahed, B. Benatallah, A. Bouguettaya and A. Elmagarmid, « Webbis: An Infrastructure For Agile Integration of Web Services », International Journal of Cooperative Information Systems, vol. 13, pp. 121-158, 2004.
- [14] S. Sattanathan, P. Thiran, Z. Maamar and D. Benslimane, « Engineering communities of web services », iiWAS Austrian Computer Society, pp. 57-66, 2007.
- [15] Z. Maamar, S. Subramanian, P. Thiran, D. Benslimane and J. Bentahar, « An approach to engineer communities of Web services concepts, architecture, operation, and deployment », IJEER, vol. 5, no. 4, pp. 1–21, 2009.
- [16] Z. Maamar, P. Thiran and J. Bentahar, « Web Services Communities: From Intra-Community Competition to Inter-Community Competition », E-Business Application for Product Development and Competitive Growth: Emerging Technologies, IGI Global, pp. 333-343, 2010.
- [17] S. Elnaffar, Z. Maamar, H. Yahyaoui, J. Bentahar and Ph. Thiran, « Reputation of communities of web services – preliminary investigation », 22nd International Conference on Advanced Information Networking and Applications-Workshops (AINA), pp. 1603-1608, 2008.
- [18] J. Bentahar, Z. Maamar, D. Benslimane and Ph. Thiran, « Using Argumentative Agents to Manage Communities of Web Services », In Proceedings of WAMIS, 2007.
- [19] J. Bentahar, Z. Maamar, D. Benslimane and Ph. Thiran, « An argumentation framework for communities of Web services », IEEE Intelligent Systems, vol. 22, pp. 75-83, 2007.

- [20] A. Benharref, M.-A. Serhani, S. Bouktif and J. Bentahar, « A Managerial Community of Web Services for Management of Communities of Web Services », 10th international conference on New Technologies of Distributed Systems, pp. 97-104, 2010.
- [21] J.-W. van't Klooster, P. Pawar, B.-J. van Beijnum, C. Dulawan and H. Hermens, « Perspectives on the Viable Mobile Virtual Community for Telemedicine », Encyclopedia of E-Business Development and Management in the Global Economy, IGI Global, 2010.
- [22] X. Blanc, « MDA en action : Ingénierie logicielle guidée par les modèles », Eyrolles, 1ère édition, Avril 2005.
- [23] J.-T.-E. Timm and G-C. Gannod, « A Model-Driven Approach for Specifying Semantic Web Services », In Proceedings of the IEEE Intl Conference on Web Services (ICWS'05), July 2005.
- [24] OMG, « OMG Unified Modeling Language Specification, version 1.4 », OMG document, Object Management Group, 2001.
- [25] <http://staruml.sourceforge.net/en/about.php>, consulted the 12/01/2013.