TOWARDS A UNIFIED MARKETPLACE FOR FUNCTIONALITY-BASED CLOUD SERVICE DISCOVERY*

Mohamed Sellami, Walid Gaaloul, Bruno Defude and Samir Tata
TELECOM SudParis, UMR CNRS 5157 Samovar, Evry, France

Keywords: Cloud Marketplace, Service Publication and Discovery, Cloud Registry, Cloud Communities.

Abstract: Cloud Computing brings an unparalleled shift to utility computing by providing unprecedented scalability and flexibility to IT services. Indeed, the concept of Cloud computing is linked intimately those of IaaS, PaaS, SaaS and collectively *aaS (Everything as a Service or Cloud services). This implies a service-oriented architecture where registries play an important role. Registries, which can be as many as the large number of Cloud services provider, compose the market where Cloud consumers and providers go to search and advertise Web services. With the proliferation of Cloud consumers and providers, finding an adequate Cloud service has become a complex task for a Cloud service consumer. In this paper we propose a functionality-driven clustering approach for Cloud service registries. This clustering will be helpful for selecting an adequate registry for service requesters. Since registries and communities are dynamic by nature, we also propose an approach for managing registries and communities to reconcile conflicts resulting from the dynamic change aspect of service registries. Experimental evaluation shows that our approach is usable in realistic situations.

1 INTRODUCTION

Cloud Computing is a new supplement, consumption, and delivery model for IT services based on Internet protocols. More and more companies are adopting the new economic model offered by Cloud Computing. For instance, in a Mckinsey Quarterly survey (The Mckinsey Quarterly, 2010) conducted in 2010 on 332 companies, 75% believe that the use of Cloud computing could drive value at their companies. Among these companies, 68% says that they are currently adopting the Cloud to set up electronic collaboration and 82% are planning to do it in the 18 coming months.

In the Cloud, services are the basis for achieving electronic B2B transactions. For worries of competitiveness and rentability, more and more companies are using Web services to adapt to new demands and expectations (Cliquet, 2011) by modeling resources as Cloud services. In our work, we are interested in Cloud service discovery. Finding a specific service in a multi-Cloud service providers environment can be a cumbersome task (for a Cloud service consumer), due to the lack of a standardized model for service discovery in this context. In fact, supposing that each Cloud provider owns its repositories of offered services, and considering that each Cloud has its own access policy to these repositories, screening all these services repositories, to discover a service, will be inconceivable for a Cloud consumer and would dramatically slow down the discovery process.

To face this issue, we propose in this paper to build an overlay marketplace over the existing physical services registries (i.e. repositories of Cloud services) belonging to different Cloud providers. To address the likely large number of service registries in such a marketplace and their poorly organized network, we propose to organize registries into communities according to the functionalities of the services they advertise. This organization allows an efficient services requester's query routing to potential registries offering services with the needed functionality. We use communities as a means for a functionality-driven organization of a distributed Cloud registry environment. So, existing registries belonging to different Cloud providers can be structured into communities and each registry belongs to at least one community with a certain extent. The registries organization inside this marketplace is built automatically, based on the functional descriptions of the services offered by a registry, and do not require any explicit intervention from Cloud providers.

In addition, registries operate within a very dy-

*This work was partially funded by the French ANR project PAIRSE.

Sellami M., Gaaloul W., Defude B. and Tata S..
TOWARDS A UNIFIED MARKETPLACE FOR FUNCTIONALITY-BASED CLOUD SERVICE DISCOVERY.
DOI: 10.5220/0003931402520257
In *Proceedings of the 2nd International Conference on Cloud Computing and Services Science* (CLOSER-2012), pages 252-257
ISBN: 978-989-8565-05-1

namic environment where changes are frequent. In fact, a new registry can join a Cloud and others can leave at any time. To ensure the evolution of the internal organization of our marketplace, we define management operations to handle (i) new registry emergence, and (ii) existing registry disappearing.

The paper is organized as follows. Section 2 introduces the proposed architecture for our marketplace. In Section 3, we present our automatic and implicit approach for organizing Cloud services registries as communities. Section 4 presents the defined management operations for handling the evolution of the created organization. The implementation efforts are shown in Section 5. Related works are discussed in Section 6. We conclude this paper in Section 7.

2 OVERVIEW OF THE PROPOSED MARKETPLACE

As defined by NIST (the National Institute of Standards and Technology), Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources defined, advertised and consumed as services that can be rapidly provisioned, released, discovered with minimal management effort or service provider or consumer interaction. Indeed, the concept of Cloud computing is linked intimately those of IaaS, PaaS, SaaS and collectively *aaS (Everything as a Service or Cloud services[2]). This implies a service-oriented architecture where registries play an important role. Cloud computing has particular characteristics that distinguish it from classical service oriented environments: (1) currently the major Cloud providers had already invested in large scale infrastructure ;(2) however little attention is given to Cloud service organization and discovery to exploit them; (3) as a consequence the Cloud service registries are poorly organized heterogeneous, and without agreed and federated interfaces; (4) and there are concerns over service availability and provision.

In this paper we describe a Cloud marketplace, which represents a portal for existing registries in different Cloud network, in order to handle the above challenges. As depicted in Figure 1, the marketplace contains an overlay network (a P2P network for example) over the physical network. Each node of the overlay will act as a proxy for a Cloud service registry in the physical network. This overlay is organized as communities and each community represents a set of registries offering services providing similar functionalities. The general architecture of a unified Cloud marketplace contains two main components: the *Marketplace Builder and Manager* and the *Cloud services Discovery System*. Just like countries' electric networks are interconnected to facilitate market trading across infrastructures and thus reduce the risks of shortage and increase efficiency, the service registry providers network would benefit from a federated organization.

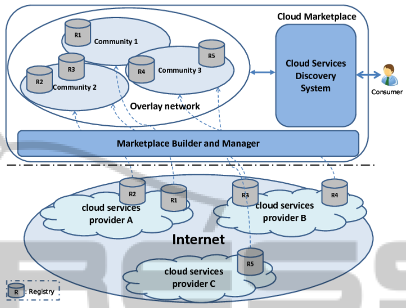

Figure 1: Overview of the proposed marketplace.

The *Marketplace Builder and Manager* is used to organize registries as communities, to deploy the overlay network and manage the organizational evolution of the overlay network. In our work, we propose to ensure this organization through a dynamic clustering technique (see Section 3). So, the organization of the different registries will be done automatically and does not require any intervention from the Cloud registry providers. The *Marketplace Builder and Manager* component also ensures the evolution of the registries organization in the view of the different possible changes in the network (see Section 4). The *Cloud services Discovery System* component implements the Cloud service discovery process. In our work, we consider Cloud service discovery as a two successive steps process: (i) discovering the adequate Cloud service registries and (ii) discovering the adequate Cloud service services inside these registries. In the rest of this paper, we will focus on the first component, the *Marketplace Builder and Manager*, of our proposed marketplace due to space limitations. Details on our work on the *Cloud services Discovery System* will be given in further papers.

3 ORGANIZING REGISTRIES IN THE MARKETPLACE

To organize registries into communities according to their functionalities[3], we associate a WSRD descrip-

[2]We confuse the terms *aaS and Cloud service

[3]Functionalities of a services registry and functionalities of the services it advertise are interchangeably used.

tion to each registry and we use these descriptions as criterion for the organization. A registry's WSRD description is automatically computed based on the services functional descriptions advertised by a registry and do not require any explicit intervention from a registry provider. Hence a WSRD description gives an overview of the functional properties of the services offered by a registry and his syntax is inspired by the WSDL format. As we provide a description of a registry (not of a service), we are only interested in the abstract section of the services description. In this work, we choose to use semantic Web service descriptions written in SAWSDL (Lausen and Farrell, 2007) to describe Cloud services. Other semantic languages, such as OWL-S (Martin et al., 2004) or WSMO (De Bruijn et al., 2005) can be also adopted. Details about the WSRD computing process can be found in our previous paper (Sellami et al., 2011).

A community of service registries will bring together registries offering similar functionalities. Since a service registry generally offer services proposing different functionalities, it is difficult to properly define in advance classes categorizing the functionalities of the different registries. To organize service registries into communities, we use a **clustering** technique (where the different communities will be deduced from the registry descriptions) rather than a classification technique (where the different communities have to be defined in advance). When using a dynamic clustering technique, the different clusters (i.e. the service registries communities) will be identified from the given input data (i.e. the WSRD descriptions) and each data point (i.e. services registry) will be assigned to one community.

Since a registry can belong to different communities at the same time, we use an overlapping clustering method, also known as fuzzy clustering, to organize our distributed registries into communities. Concretely, we proceed in three steps to build registry communities:

1. Step 1: WSRD descriptions are processed to map to the term-document structure.
2. Step 2: we propose a method to measure the distance between two registry vectors as needed by the fuzzy C-means Algorithm.
3. Step 3: using the results of the two previous steps we apply a fuzzy clustering technique to build registry communities.

The reader is referred to (Sellami et al., 2011) for details on our clustering approach. After these three steps, we get C communities represented by their centroids. These centroids represent the mean functionality f of these communities. In addition, to each community c_j are associated N membership vectors $U_j(r_i), i = 1 \dots N$ indicating the degrees of membership of the N registries to the community. These data allow us to infer the membership degrees MEM of the N registries to organize into communities. However, using this technique, the membership degree of some registries to certain communities may be very low. For this, we define a threshold th for membership degrees. If the membership degree of a registry to some community is lower than this threshold, it will not be considered as a member.

Formally, we define the obtained **communities** as couples $c = (f, M)$ where: f is a vector representing the mean functionality of the community c and M is the set of community *members* (i.e. registries).

Registries inside a community are defined as couples $r = (f, MEM)$ where: f is a vector representing functionalities offered by the advertised Web services within the registry and MEM represents the registry membership degrees to the different communities in the network. MEM is represented as a binary relation defined as follows. $MEM = \{(c,d) | c \in C, d \in [0,1]\}$ where: C is the community set and d is the membership degree of the registry r to the community c.

4 MANAGING THE REGISTRIES ORGANIZATION IN THE MARKETPLACE

The obtained organization in the previous section will be used to deploy the overlay network (a P2P network for example) of our marketplace. Each node of the overlay will act as a proxy for a Cloud services registry and the nodes will be organized in communities according to the results given by our clustering technique. Technical details are given in Section 5.

However, in Cloud environments it is normal that new registries may emerge and existing registries may disappear for some reason. To maintain the organization of the different registries according to their functionalities and to guarantee its evolution after these changes, we define operations for handling the following situations: Cloud services registries joining the marketplace (Section 4.1) and Cloud services registries leaving the marketplace (Section 4.2).

4.1 Cloud Services Registries Joining the Marketplace

When a new registry r is identified, first its WSRD description is computed. After that, the registry will be guided to the adequate communities according to its set of membership degrees MEM. MEM is computed by the *SelectCommunity* algorithm (Algorithm 1). This algorithm takes as input the current registry's WSRD description. The membership degree is computed using the function U[4]. This algorithm outputs the set of membership MEM containing the membership degrees of the current registry to the different communities in the marketplace. Taking into account that the membership degree must be greater than a certain threshold `th` (line 4), the registry will be guided to the adequate communities ((line 5)).

Algorithm 1: SelectCommunity.

Require: r :registry
Ensure: $r.MEM$
1: **for each** community $c \in C$ **do**
2:     $m \longleftarrow U_i(r)$
3:     $r.MEM \longleftarrow r.MEM \cup \{(c,m)\}$
4:     **if** $m > \mathtt{th}$ **then**
5:         $M \longleftarrow M \cup \{r\}$
6:     **end if**
7: **end for**

When the membership degrees of a registry r to all the existing communities are lower than the threshold `th`, it provides necessarily a new functionality inexistent in the network. In this case, a new community c_{new} is established automatically. The community mean functionality $c_{new}.f$ will be the same as the functionality $r.f$ proposed by the registry that triggered the community creation. Afterwards, some *members* are recruited for the new community. In this aim, the membership degrees of existing registries to the new community are computed. Every registry having a membership degree higher than the threshold `th` will be a member of c_{new}. The community creation as well as the necessary precondition and effects are displayed in Algorithm 2.

Algorithm 2: CreateCommunity.

Require: r: the registry that triggers the community creation
Ensure: c_{new}: the new community
1: ***precondition***: $\forall d \in ran(r.MEM), d < \mathtt{th}$
2: Community c_{new};
3: List members;
4: $c_{new}.f = r.f$
5: **for all** communities $c \in C$ **do**
6:     **for all** registries $r_c \in c.M$ **do**
7:         $m \longleftarrow U_{new}(r_c)$
8:         $r_c.MEM \longleftarrow r_c.MEM \cup (c_{new}, m)$
9:         **if** $m > \mathtt{th}$ **then**
10:             $members \longleftarrow members \cup \{r_c\}$
11:         **end if**
12:     **end for**
13: **end for**
14: ***effect***: $c_{new} \in C \wedge c_{new}.f = r.f \wedge c_{new}.M = members$
15: **return** c_{new}

[4]The same function U is also used in our community building approach (Sellami et al., 2011)

Also, when a registry r is assigned to a community c, c may become sparse and thus may need to be split into two or more sub-communities. The community sparsity describes a non density in the center vicinity and a dispersion between members. In this case, the Mcut method (Ding et al., 2001; Nie et al., 2010) can be applied for splitting a community into several sub-communities. Concretely, this method allows to minimize the similarity between two sub-communities, while maximizing the similarity within each sub-community (details on how to split communities can be found in (Bouchaala et al., 2011)).

4.2 Cloud Services Registries Leaving the Marketplace

When an existing registry r leaves the Cloud or is unavailable for some reason, its representing node in the overlay network is to be removed from the marketplace. In this case, if a community c contains this registry, r has to be removed from it member set $c.M$. If c does not contain any registry after removing r ($c.M = \{\}$), c will be automatically dismantled. Otherwise, we check whether c can be merged with another community in the overlay network. If c cannot be merged, the mean functionality f of c will be updated. Finally, the membership degree of each registry in the different communities is recomputed.

Algorithm 3: MergeCommunity.

Require: c: a community
Ensure: **true** or **false** depending on whether c is merged with another community or not
1: $isMerged \leftarrow$ **false**
2: **for all** $c_{oth} \in C \setminus \{c\}$ **do**
3:     **if** $c.M \nsubseteq c_{oth}.M$ **then**
4:         **continue**
5:     **end if**
6:     $isMerged \leftarrow$ **true**
7:     $C \leftarrow C \setminus \{c\}$
8:     **break**
9: **end for**
10: **return** $isMerged$

Next we introduce the *MergeCommunity* algorithm. c represents the community to be merged. As presented in Algorithm 3, if the members of c represent a subset of the members of another community (i.e., c_{oth}), c is removed from the overlay network (lines 6-12).

5 IMPLEMENTATIONS

We have prototyped our approach in java. Cloud services are described in a format like SAWSDL. We created a test collection of semantic Web service descriptions written in SAWSDL. These service descriptions are annotated using concepts from an ontology described in OWL. For our experiments, we generated a collection of 1400 SAWSDL descriptions. We split up the generated SAWSDL descriptions into 7 Web service registries $r_j, j = 1 \dots 7$. To organize these registries into communities, we calculate for each registry its WSRD description. We developed a tool, *WSRDCreator*, to create WSRD descriptions from a set of SAWSDL descriptions. *WSRDCreator* generates a vector representation of each registry's created WSRD description. This vector $r_j.f$ represents the functionality f offered by a registry r_j and serves as input for our community building approach. We use these WSRD vector representations to build the communities of registries. For this, we developed *CommunityBuilder* a tool implementing our community building approach. Using *CommunityBuilder*, we construct 4 communities for the 7 registries.

Based on this organization, we now deploy the overlay network of our marketplace. The JXTA P2P platform[5] is suitable to simulate an overlay network organized as communities above the physical registries network that we have put in place at the previous steps. A registry community is then viewed as a *Peer Group*[6] where each service registry is mandated by a JXTA peer. We developed a *CommunityConsole* to deploy our test-bed on the JXTA platform.

To test the feasibility of our registries organization management approach, we implemented a *Community Manager*. This tool graphically simulates a network of registries organized as communities and implements our algorithms for managing the registries organization in the marketplace. We used this tool to simulate a network of communities of registries and introduce changes to test our managing operations.

6 RELATED WORKS

In this paper, we presented our approach for building a unified marketplace for Cloud services by virtually organizing Cloud service registries as registry communities. We also specified the needed management mechanisms to guarantee the evolution of a community organization. As far as we know, this aspect has not been studied before in Cloud environments. Thus we provide in this section a state of the art study of Web services registries organization in service oriented environments. Several Web services discovery systems rely on a distributed registry environment to overcome the problems related to single centralized registry based discovery (bottlenecks, single point of failure ...). Our literature review yielded a good number of research projects that used distributed registries such as (Sivashanmugam et al., 2004; Pilioura and Tsalgatidou, 2009; Ayorak and Bener, 2007; Du et al., 2005), but as far as we know, none of them have promoted the idea of using a functionality based semantic description model to organize Web service registries.

MWSDI (Sivashanmugam et al., 2004) propose a distributed Web services registry infrastructure for Web services discovery. In this work, the registry network is structured into federations using a specialized ontology called the *Registries Ontology*. This ontology allows grouping different Web service registries based on their business domain as the authors map each registry to one or several nodes in the *Registries Ontology*. Associating with each registry a specific domain or a group of domains from the *Registries Ontology* enhances Web services discovery. In (Pilioura and Tsalgatidou, 2009), the authors also propose a distributed registry infrastructure called PYRAMID-S, where Web service registries are categorized by concepts token from a specialized ontology.

In (Ayorak and Bener, 2007), an hybrid architecture for Web services discovery in distributed registry environment is proposed. Unlike previous approaches, the structure of the network is build based on the business domain of the services advertised by the registries: services operating in similar domains are placed in the same group.

Authors in (Du et al., 2005) present Ad-UDDI; an active and distributed infrastructure for organizing a network of UDDI registries. The proposed infrastructure is formed by three layers: The *root registry layer* contains an Ad-UDDI registry containing information about the registries forming the network of registries, the *business service registry layer* contains the network of registries and the *service layer* contains the services advertised by these registries. Authors organize statically their *business service registry layer* by using a taxonomy (GICS[7]) classifying different real industrial sectors, industries, sub-industries and industry groups.

To sum up, our literature review yielded a good number of research projects that used distributed registries, but as far as we know, none of them have promoted the idea of using a functionality-based semantic description model to organize Web service registries. Compared to the above literature review, our contributions can be summarized in three main points:

1. Our registry description and organization is based only on **implicit knowledge** using existing adver-

[5] http://jxta.kenai.com/

[6] *Peer Group* is a JXTA concept defined as a set of peers offering a specific service.

[7] GICS : Global Industry Classification Standard.

tised service descriptions. Thus, our approach is **self-contained** within the Cloud service discovery process, **independent** from any explicit or human centric or error prone knowledge.

2. Our approach proposes a **functionally-based** organization of service registries into communities. Such an organization enhances the services discovery process since the registries will be grouped according to the functionalities proposed by their advertised services, and thus a service requester's query can be guided to the adequate registry cluster for his needs.

3. By fuzzy clustering service registries according to their service descriptions, our approach has intrinsically the means for a **dynamic, flexible and automatic management** of service evolutions.

7 CONCLUSIONS

We presented an approach for building and managing a marketplace for Cloud services. In the proposed marketplace, the Cloud services registries are organized as communities. These communities are implicitly and automatically created using the registries WSRD descriptions. Our registries organization approach uses a functionality-driven clustering and organizes registries according to the functionalities of the Cloud services they advertise. This functionality-driven organization of the marketplace registries enhances services discovery. In fact, a service requester's **search space** can be **reduced** to the registry community advertising services offering the needed functionalities. We defined the required management operations to ensure the evolution of the registries organization. We also demonstrated and deployed the proposed approach on top of a JXTA-based distributed registries environment and implemented our management operation to test the feasibility of our management approach.

Our future research work will focus on *aaS services specification and discovery. In this work, we have used traditional Web service description in order to specify the WSRD description of a Cloud services registry. However, the Open Cloud Computing Interface (OCCI) (Metsch et al., 2010) has emerged as standard to describe *aaS services specific features. Originally initiated to create a remote management API for IaaS model based services, allowing for the development of interoperable tools for common tasks including deployment, autonomic scaling and monitoring, it now can be used to severe other *aaS services as well. As part of our perspectives, we aim in our future work to extend our approach in order to take into account the OCCI specification.

We also plan to define the *Cloud services discovery system* component to test the efficiency of our functionality driven organization of the Cloud services registries for services discovery. This discovery system will use the functional requirements of a services requester to reduce the search space by selecting adequate communities and in a second time will combine his functional and non-functional requirements for an efficient services selection.

REFERENCES

Ayorak, E. and Bener, A. B. (2007). Super peer web service discovery architecture. In *ICDE, April 15-20, 2007, Istanbul, Turkey*, pages 1360–1364.

Bouchaala, O., Sellami, M., Gaaloul, W., Tata, S., and Jmaiel, M. (2011). Modeling and managing communities of web service registries. In *WEBIST, Noordwijkerhout, The Netherlands, May 6-9, 2011*.

Cliquet, G. (2011). *Method of innovation in the era of Web 2.0*. Phd thesis, Paris Institute of Technology.

De Bruijn, J. et al. (2005). Web service modeling ontology (wsmo). Technical report.

Ding, C. H. Q., He, X., Zha, H., Gu, M., and Simon, H. D. (2001). A min-max cut algorithm for graph partitioning and data clustering. In *ICDM'01, San Jose, California, USA*.

Du, Z., Huai, J., and Liu, Y. (2005). Ad-uddi: An active and distributed service registry. In *TES, Trondheim, Norway, September 2-3, 2005, Revised Selected Papers*.

Lausen, H. and Farrell, J. (2007). Semantic annotations for WSDL and XML schema. W3C recommendation, W3C. http://www.w3.org/TR/2007/REC-sawsdl-20070828/.

Martin, D. et al. (2004). Owl-s: Semantic markup for web services. Technical report.

Metsch, T., Edmonds, A., and Bayon, V. (2010). Using cloud standards for interoperability of cloud frameworks. Technical report.

Nie, F., Ding, C., Luo, D., and Huang, H. (2010). Improved minmax cut graph clustering with nonnegative relaxation. In *ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010*.

Pilioura, T. and Tsalgatidou, A. (2009). Unified publication and discovery of semantic web services. *ACM Transactions on the Web (TWEB)*, 3(3).

Sellami, M., Gaaloul, W., and Tata, S. (2011). An Implicit Approach for Building Communities of Web Service Registries. In *iiWAS'2011, 5-7 December, 2011, Ho Chi Minh City, Vietnam*.

Sivashanmugam, K., Verma, K., and Sheth, A. P. (2004). Discovery of web services in a federated registry environment. In *ICWS'04, San Diego, California, USA*.

The Mckinsey Quarterly (2010). How IT is managing new demands: McKinsey Global Survey results.