ISSN 2229-5518

Social Networks' Interaction and Social Web Services

Mrs. Sumathi
Asst. Professor, Computer Science and Engineering, Canara Engg College, Mangalore, Karnataka, India, pawarsumathi@gmail.com

Dr. Ashok Kumar
Professor, Computer Science and Engineering, NMAM IT, Nitte, Karkala, Karnataka, India
ashok@nitte.edu.in

Abstract-- At present the discovery techniques are registry-based and rely on syntactic and sometimes semantic descriptions of Web services' interfaces. Registries have several drawbacks: for instance, syntactic discovery returns results with low precision. Web services are treated as independent elements in these registries and present registries don't record services' past interactions. To address these issues, this paper embrace social networks to record past interactions and to develop that refer to here as social Web services. This requires first building and maintaining Web services' social networks and then using them to find, for example, highly collaborative peers. Service engineers often struggle to locate the relevant Web services needed to satisfy users' requests. In this paper it is also described that how service engineers can capitalize on Web services' interactions — namely, collaboration, substitution, and competition — to build social web service application for service discovery.

Index Terms : UDDI, UPnp, WSDL, Social Networks, SOA

1. INTRODUCTION

Social networks exemplify the tremendous popularity of Web2.0 applications, which help users become proactive; colloquially, we can refer to users now as prosumers, providers and consumers at the same time. Prosumers post definitions on wikis, establish groups of interest, and share tips and advice. These various operations illustrate the principles of "I offer services that somebody else might need" and "I require services that somebody else might offer" upon which SOA(Service Oriented Architecture) is built. Service offerings and requests demonstrate perfectly how people behave in today's society, imposing a social dimension on how Web services must be handled in terms of description, discovery, binding, and composition. What if this social dimension is the missing link? It could serve as an additional ingredient to the formal methods that support SOA needs, namely, service description, discovery, binding, and composition.

[1]Weaving social elements into Web service operation means new social Web services (SWSs) that will

- Establish and maintain networks of contacts;
- Put users either explicitly or implicitly in the heart of their life cycle, enabling additional functionalities through collaboration;
- Rely on privileged contacts when needed;
- Form with other peers strong and long-lasting collaborative groups; and
- Know with whom to partner to minimalize ontology reconciliation.

For a developer developing a software system with a public web service, there are several tasks he/she needs to accomplish: finding the service, understanding the service, and integrating the service into the system. During these tasks, the descriptions of the service play an important role.

With the descriptions, the developer can understand the service; and further decide whether the service is the one he/she needs; and then learn to integrate the service into the target system.

Generally, a web service is mainly described by a WSDL (Web Service Description Language) file expressed in Web Service Description Language. For a web service, there are two types of descriptions in its WSDL file: structural descriptions, and non-structural descriptions. Structural descriptions provide syntactic declaration for the service, such as the message types, transmission protocol, and service location. Structural descriptions can help consumers learn about 'what the service looks like' but cannot show them 'what functionalities the service provides' or 'how to use the service'. Comparatively, non-structural descriptions, i.e. the textual descriptions expressed in natural language which are embedded in the wsdl:documentation element in WSDL file, can usually show consumers what the service can do and how to use the service. Compared with analyzing syntactic declarations in the WSDL file, a consumer can understand the web service better by reading and interpreting the non-structural descriptions of the service.

Through the investigation into the public web

Author is currently persuing PHd in VTU University under the guidance of Co-Author and email Id: pawarsumathi@gmail.com
Co-Author is Professor in NMAMIT, an autonomous university and also writer of many technical articles.
Email id:ashok@nitte.ac.in

services on the Internet, it is found that, in addition to services' WSDL files, there are many useful descriptions for services in the web pages that are related to the web services, such as discussions about the usage of the services, advertisements with introductions to the services. All these information is helpful for consumers to understand the service more efficiently. These descriptions are particularly important for the web services whose WSDL files do not contain enough information. Moreover, it has been accepted that for an entity on the Internet, there might be descriptions about it in its related web pages. Even though these descriptions are useful for service discovery it is not sufficient to satisfy prosumer's needs. Therefore it is necessary to develop social web service applications on the basis of social network interaction that is discussed in this paper.

2. UDDI NOMENCLATURE

[2]UDDI (Universal Directory Description and Interchange) is a specification that helps providers in registering their service and thereby advertises the product as well as service information. UDDI services help requester to find the service and all the related details. Provider could be using a software tool that conforms to the specification for publishing and updating business and service-related information in the registry. On the other hand the requester could be using a software tool / Integrated Development Environment(IDE) / browser to query and obtain required business and service-related information.
UDDI Browser registries (UBR) are amenable to storing new business information, editing or modifying existing information or querying for business information using one or more key words. The UBRs are global, public online directories. These UBRs offer services to business for publishing the company data as well as the services-related data for public and global use.
The UDDI specification uses some specific terminology that needs to be understood before exploring the details. They are Node API Sets, UDDI Node, UDDI Registry, Information Model and Data Structure.

2.1 Node API Sets

These are API sets that are used for inter-UDDI communication implementations. The purpose is to manipulate the data stored in different UDDI implementations. The node API sets are available for implementing a variety of tasks such as UDDI Inquiry, UDDI publication, UDDI subscription, and UDDI Security.

2.1.1 UDDI Node

A system that hosts an application to support at least one of the Node API sets is called a UDDI Node. A UDDI node must ensure the following :

- Interaction with the UDDI data through API sets.
- Membership with exactly one Registry.
- Ability to access and manipulate UDDI data.

2.1.2 Public and Private Registries

Many leading organizations are already offering registry services. The public registries that are in conformance with the UDDI specifications are set up by organizations such as IBM, Microsoft, XMethods etc.

Private business registries are also available, however for use among a group of selected companies who are agreeing partners. However the specifications of these private registries may not be in full conformance with the UDDI specifications. The UDDI specification represents data in six different types of data structures. These data structures are referred to as entities. They are BusinessEntity, BusinessService, BindingTemplate, tModel, publisherAssertion, subscription. These data structures represent the core of the UDDI information model.

3. DRAWBACK OF EXISTING SYSTEM

In the discovery stream, existing standards such as UDDI let enterprises publish their Web services so they can be identified using keyword-based searches. As UDDIs proliferate, screening them all has become time consuming and ineffective.
Other alternatives propose information-retrieval techniques such as term-frequency / inverse-document frequency to identify the operations of a Web service that are most similar to the user's description. Unfortunately, these techniques don't reveal the semantic relationships between Web services. In the Semantic Web community, description logic is used to develop inference mechanisms that establish similarity among Web services.
Unfortunately, all these discovery approaches handle Web services as isolated components that don't interact with each other. It is also noted that although Semantic Web services technologies have shown their benefits, socio-economic aspects

devoted to service creation and annotation are missing. This drawback in existing system creates the need of social networks for development of social web services.

4. Literature Survey

In the social computing stream, the combination of Web services and social networks is to a certain extent new, so several research opportunities are still untapped. Abderrahmane Maaradji and his colleagues suggest a social composer that relies on interactions between users to advise future users on which Web services to select[3]. Trust among service providers, service consumers, and a service themselves is the social element that this composition takes into account.

In the domain of Web services, Benatallah et al. define community as a collection of Web services with a common functionality although these Web services have distinct non-functional properties [4].

Medjahed and Bouguettaya use community to organize Web services that share the same domain of interest with respect to a certain ontology [5].

Maamar et al. define community through the functionality of a representative abstract Web service (called master) that leads the community, without explicitly referring to the concrete Web services (called slaves) that exist in this community and implement this functionality at run-time [6].

Wan et al. define communities of Web services as virtual spaces that can dynamically gather different Web services having complementary or related functionalities [7].

On one hand, social computing is the computational facilitation of social studies and human social dynamics as well as the design and use of information and communication technologies that consider social context [8].

Social computing is also about collective actions, content sharing, and information dissemination in general. On the other hand, service-oriented computing builds applications on the principles of service offer and request, loose coupling, and cross organization flow[9].

Collaboration, substitution, and competition are some of the links that can connect Web services together. It is mentioned earlier the works of Maaradji et al. and Xie et al. Maaradji et al. propose a social composer that advises on the next actions to take in response to events such as Web services selection[10].

Xie et al. introduce a framework for semantic service composition based on social networks [11]. Wu et al. rank Web services using non-functional properties and invocation requests at run-time. A Web service's popularity as analysed by users is the social element that is considered during ranking [12]. Tan et al. apply social networks analysis to mine and analyse a workflow repository, focusing on service usage patterns [13].

Last but not least, Nam Ko et al. [14] discuss the social Web in which a new type of services called social-networks connect services help third party develop social applications without having to build social networks. All these works aim at providing support to users engaged in social networks by using services.

5. Community and Social Networks

A community is "a group of people living together and/or united by shared interests, religion, nationality, etc." (Longman Dictionary).

Some direct benefits out of the adoption of communities include:

· Communities constitute pockets of expertise since they gather Web services with similar functionalities in the same "space". The search for required Web services is narrowed down to a limited number of specific communities.

· Communities smoothen Web services substitution in case of failure. Potential substitutes are already known and thus directly contacted to check their willingness to act as substitutes.

· Communities can be structured internally using different models such as slave-master, peer-to-peer, or mixture of all of these.

5.1 Types of relationships

As stated earlier different types of relationships can be detected within and across communities. Table 1 summarizes them and present their respective purposes and participating Web services[15].

· Supervision relationship between the master Web service and the slave Web services.

· Competition relationship between the slave Web services since they all offer the same functionality and only one slave Web service is selected at a time to satisfy a user's request.

· Substitution relationship between the slave Web services since they all offer the same functionality, so they can replace each other.

· Collaboration relationship between the slave Web services engaged in the same compositions.

· Recommendation relationship between the slave Web services, through their respective master Web services, so that a high compatibility level between the slaves Web services is reached when compositions are built.

Thus these relationships across communities provides information for Web services interactions by which already discovered services can be used whenever it is required to satisfy prosumer's needs.

TABLE 1. RELATIONSHIPS WITHIN AND ACROSS COMMUNITIES

Type of Social Relationship	Community		Participating Web Services	Purpose
	Intra	Inter		
Supervision	✓		Master - Slave	Users requests' satisfaction
Competition	✓		Slave - Slave	Web services enhancement
Substitution	✓		Slave - Slave	Web services high availability
Collaboration		✓	Slave - Slave	Composition post-assessment
Recommendation		✓	Slave - Slave	Composition compatibility

6. Discovery between multiple home social networks across the Web

Service discovery protocols allow applications to locate all available instances of a particular type of service in closed environments, and without prior configuration. There are several architectures aimed at service discovery within a home network, for instance, UPnP can better justify why interoperability among networked home appliances should adopt a service-oriented approach rather than a hardware oriented approach.

[16]This section proposes an overview of UPnP framework for service discovery between multiple home social networks across the Web. This discussion focuses on the management scheme for UPnP devices, which uses the standard UPnP device network protocol. This system adopts a multiple system management center (SMC) server into a gateway, which deploys a proxy map to remote devices on a local network using the gateway's IP address, collects all services within the local home network, and exchanges a list of local services with other gateways of remote home networks. Whenever a local device queries specific services, the gateway will response on behalf of the remote devices that have the requested services will be initialization update message in device manager.

A standard UPnP protocol stack is as show in Figure 1.

6.1 Discovery(device and service discovery)

Simple Service Discovery Protocol (SSDP), which uses the connectionless UDP's HTTPU and HTTPMU, as based on implementation of the agreement rather than a connection-oriented TCP protocol. When a Control Point (CP) is added, SSDP packets can be sent to the network, by searching whether (Search) UPnP devices on the network are Device Control Protocol.

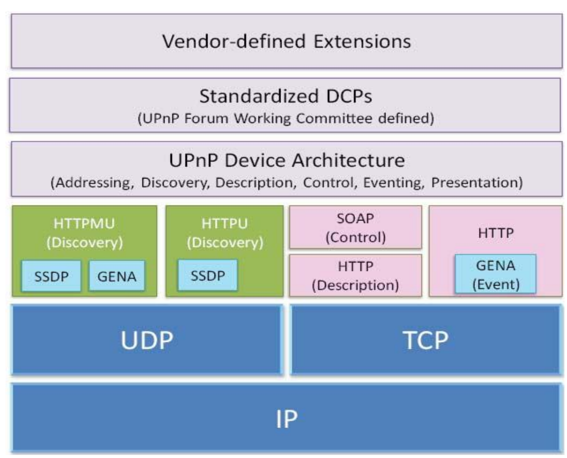

Figure 1. UPnP protocol stack

When a Device Control Protocol (DCP) is added, SSDP can send packets to the network, which publicity (Advertise) tells all the new devices that CP added. If CP cannot find other

TABLE 2. PAST INTERACTIONS OF USERS

Time	PERSON	SERVICE1	SERVICE2	…SERVICE N
T1	P1	Weather Forecast	Translator	Send mail
T2	P2	Weather Forecast	Translator	……
T3	P3	Check Eligibility	CreateBankAccount	……
T4	P4	Check Eligibility	DoEcommerceTransaction	……
T5	P5	Weather Forecast	Translator	Send mail

devices, as it is not required to respond to Search and will not take the initiative to Advertise; while DCP cannot take the initiative to Search, only active or passive responses to the CP Advertise Discovery requirements.

The descriptive nature of SOAP renders it the fastest service device protocol, as service descriptions are sent by multicast in a simple string, while the UPnP processes complex XML documents. It should be noted that the response time for UPnP discovery requests are exceptionally high as shown in the figure 2 and can be attributed to thread and timeout management during implementation of the stack.

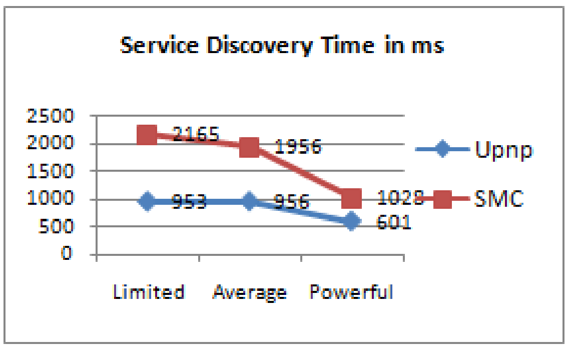

Figure 2. Service Discovery time in miliseconds

This study implements an SMC user interface to access and control a home network of UPnP devices. "Error! Reference source not found" presents the service measurements for discovering UPnP devices and SOAP service access (Broadcast-based), which uses a Widget-based UPnP framework (First service description to be returned is time). From the measurements presented in Error! Reference source not found it can observe the problem of evaluating other UPnP stacks in the same service access update.

However, Web Services technology is not suitable for home networking. First are the discovery and configuration process for UDDI, the service discovery protocol for Web Services, which can involve programmers and administrators. Home networks are complex and not easily used, thus they should maintain simple and automatic processes for services of discovery; Secondly, Web Services require system management, e.g. a services directory applied to home networking, this study does not propose using any single centralized network management method that it is not specifically recommended for centralized management in home network applications.

7. BUILDING SOCIAL WEB SERVICES' APPLICATIONS

Social network interactions can be used as knowledgebase of discovery model. In the social networks like facebooks the timeline gives past and current interactions done by different users. This can be extracted and stored in the database which can be used as training set for generating rule.

Table 2 gives some records of past interactions of user with different services in the social networks. This data set is retrieved from a social network and includes user interests. If all the data sets retrieved for a period of long time, it can be used as training set of knowledge base model.

Association analysis and confidence of the rule set gives the probability of selection of this web service in the future.

{Weather Forecast, Translator} : Frequency 3/5

{Weather Forecast, Translator, Send Mail} : Frequency 2/5

By using this set of associated items following rule will be generated.

Rule R1 : service1="Weather Forecast" ∩ service2="Translator" → Send mail

R2 : Weather Forecast → Translator.

The confidence of the Rule R1 is 2/3 = 0.66 = 66%

Confidence of the rule R2 is 3/3 = 1 = 100%

A decision model based on this rule set will be developed and based on this decision model necessary web services will be composed and social web service application will be build. Web services which are included in these applications are already used by the users and can be used in the future confidently. The social web services application also saves the time of composing web services for the future web service requests.

The decision model developed using above model forms the static knowledgebase. Even though this is useful for future selection of web services, this also requires a model which presents the user interests in different domain. For this purpose information from domain experts will be collected and composition model will be build which will highly satisfies the user requests.

Let's illustrate the use of Web services social networks with a scenario related to purchase orders. Initially, a customer places an order for products via CustomerWS. Based on this order, CustomerWS obtains details on the customer's purchase history from CRMWS (that is, customer relationship management). Then, CustomerWS forwards these details to BillingWS, which calculates the customer's bill based on this history (for example, discount eligibility) and then sends it to CRMWS. The lateer prepares the detailed purchase order and sends it to InventoryWS for order completion. For in-stock products, InventoryWS sends a shipment request to ShipperWS to deliver the products to the customer.

Sometimes this delivery is revisited as per weather conditions returned by WeatherWS. For out-of-stock products, InventoryWS sends a supply message to the selected SupplierWS, which provides ShipperWS with the products for subsequent shipments to the customer.

If these interactions are saved then these sequence of web service operations will be used as knowledge base for future selections using data mining techniques as shown in the rule based analysis.

When a social network application is build for the first time, Web services grouped into three clusters according to their degrees of similarity(DS) with the Web service root: weak (for example, $0 < DS \leq 0.33$), average ($0.33 < DS \leq 0.66$), and strong ($0.66 < DS \leq 1$). Similarity degree will be used later to assign initial edge weight values. If a peer has a strong similarity degree with the root, then place it into the root's strong similarity cluster. The same placement process applies for average and weak-similarity degrees, and continues as long as services become available and agree to be part of the social network. While clustering is in progress, Web Services are connecting within the social networks as well, which ultimately extends these networks.

8. Conclusion

In response to challenges that today's enterprises face like competition and market volatility, it is largely recommended that these enterprises' business processes should be loosely coupled and capable to cross organization boundaries. Web services are among the technologies that implement such processes, in addition to their capacity to work together through composition. A rapid literature review on composition approaches suggests an exhaustive list of works on different issues related to discovery, semantics, security, etc.. Unfortunately the interactions that arise between Web services before and after they engage in compositions are somehow overlooked. Web Services collaborate when they offer complementary functionalities and compete when they offer similar functionalities. Future works will establish an ontological model of UPnP devices, such as, an ontology-based representation of UPnP devices and services for dynamic context-aware computing applications.

Acknowledgment

The Co-Author I wish to thank for his great encouragement and support.

References

[1] Why Web Services Need Social Networks Zakaria Maamar Zayed University, Dubai , Hakim Hacid, Alcatel-Lucent Bell Labs, Paris , Michael N. Huhns University of South Carolina

[2] B.V. Kumar, S.V. Subrahmanya "Web Services An Introduction" Tata McGrawHill Publication, Page Number 119-131.

[3] A. Maaradji et al., "Towards a Social Network Based Approach for Services Composition," Proc. Int'l Conf. Comm. (ICC), IEEE Press, 2010, pp. 1-5.

[4] B. Benatallah, Q. Z. Sheng, and M. Dumas. The Self-ServEnvironment for Web Services Composition. IEEE Internet Computing, 7(1), January / February 2003.

[5] B. Medjahed and A. Bouguettaya. A Dynamic Foundational rchitecture for Semantic Web Services. Distributed and Parallel Databases, Kluwer Academic Publishers, 17(2), March 2005.

[6] Z. Maamar, S. Subramanian, P. Thiran, D. Benslimane, andJ. Bentahar. An Approach to Engineer Communities of Web Services - Concepts, Architecture, Operation,and Deployment-International Journal of E-Business Research, IGI Global Publishing, 5(4), 2009.

[7] C. Petrie and C. Bussler. The Myth of Open Web Services - The Rise of Service Parks. IEEE Internet Computing, 12(3),May / June 2008.

[8]Z. Maamar, S. Subramanian, P. Thiran, D. Benslimane, and

ISSN 2229-5518

J. Bentahar. An Approach to Engineer Communities of Web Services - Concepts, Architecture, Operation, and Deployment -. International Journal of E-Business Research, IGI Global Publishing, 5(4), 2009.

[9] F.Y. Wang et al., "Social Computing: From Social Informatics to Social Intelligence," IEEE Intelligent Systems, vol. 22, no. 2, 2007, pp. 79-83.

[10] A. Maaradji, H. Hacid, J. Daigremont, and N. Crespi. Towards a Social Network Based Approach for Services Composition. In Proceedings of the 2010 IEEE International Conference on Communications (ICC'2010), Cap Town, South Africa, 2010.

[11]X. Xie, B. Du, and Z. Zhang. Semantic Service Composition based on Social Network. In Proceedings of the 17th International World Wide Web Conference (WWW'2008), Beijing, China, 2008.

[12] Q. Wu, A. Iyengar, R. Subramanian, I. Rouvellou, I. Silva Lepe, and T. Mikalsen. Combining Quality of Service and Social Information for Ranking Services. In Proceedings of ServiceWave 2009 Workshops held in conjunction with the 7th International Conference on Service Service-Oriented Computing(ICSOC'2009), Stockholm, Sweden, 2009.

[13] W. Tan, J. Zhang, and I. Foster. Network Analysis of Scientific Workflows: A Gateway to Reuse. IEEE Computer, 43(9), September 2010.

[14] M. Nam Ko, G. P. Cheek, M. Shehab, and R. Sandhu. Social Networks Connect Services. IEEE Computer, 43(8), August 2010.

[15] Zakaria Maamar Zayed University Dubai, U.A.E, Hamdi Yahyaoui Erbin Lim, Kuwait University and Philippe Thiran PReCISE Research Center, Namur University, Safat, State of Kuwait "Social Engineering of Communities of Web Services" 2011 IEEE/IPSJ International Symposium on Applications and the Internet pp. 100-109.

[16] Widget-Based Framework for Web Service Discovery on Multiple Home Social Network Chih-Wei Hsu, Sheng-Tzong Cheng Dept. of computer science and information engineering National Cheng Kung University, Tainan, Taiwan, Chen-Fei Chen Department of Information and Communications, Shih Hsin University, Taipei, Taiwan.