

Interoperability Issue in web services and Multi-Agent Systems

Juwel Rana

juwel.rana@dit.unitn.it

Department of Information and Communication Technology (DIT)
University of Trento, Trento, Italy

Abstract: *Service orientation is a way of thinking in terms of services and service-based development and the outcomes of services. Web services, business processes are part of this architecture. Agent based services are developed based on distributed platforms by composing autonomous entities, called Agents. In this paper we will present different ideas considering architectural issues for web services and multi-agent systems to provide higher level of inter-operability by providing a common framework to work together. In this paper we haven't classify a service as agent service or web service. Here I consider a service as a unit of task which can be an instance of agent service or a web service and these services are publishable, findable and accessible in a uniform way. We have proposed different ideas for providing middleware to solve inter-operability issue between web services and multi-agent systems.*

1. Background

A Web service is a software application identified by a URI, whose interfaces and binding are capable of being defined, described, and discovered by XML artifacts, and supports direct interactions with other software applications using XML-based messages via Internet-based protocols. A network component in Web Services architecture can play one or more fundamental roles which are Service Provider, Service Requester and Service Broker. The three fundamental operations of Web Services are Publish Find and Bind. For instance Google Calendar is a web service which is getting popular in for personal and commercial usages.

A Multi-Agent System is a distributed system composed of autonomous entities, called agents. These agents need to interact and cooperate in order to achieve global tasks. The tasks are defined by the agent owner on behalf agent works. One of the main properties of MAS is that it relies rather on the distribution of cooperation algorithms than on centralized processes. The decentralized and loosely coupled nature of the network makes it possible to design applications that are highly flexible, scalable and adaptive.

2. Convergences between Web Services and Agent Services

In Service Perspective, we find the following similarities issues between web services and multi-agent systems:

1. Web services and multi-agent systems are software application
2. Web services are developed based on distributed platform where-as multi-agent system can be developed same platform.
3. Web services and agent services have been standardized by different federations. W3C [4] provides standardization for web services whereas FIPA [3] provides standardization for agent services.

4. These services are self publishable. It means that these services are published in some kind of service repository (UDDI, DF-Agent) to provide the port types of these services.
5. At the low level communication both of these technologies uses the same protocols like http and smtp.

In methodological perspective, starting phase is to build a web service to design and implement the service. Then test feasibility, usability, applicability of the service and with satisfactory results deploys and executes the service. The last important is to manage the service which are related with Service-Level Agreement (SLA) Management Auditing, Monitoring, and Troubleshooting.

For Multi-Agent systems, the generic methodology [6] could be start from initial system specification and define roles of particular agents. Then design use cases of the specific operations and refine the roles. After coming in a satisfactory stage, the next step is to create agent classes, constructing conversations and finally design system and implementation it and deply it in agent platform.

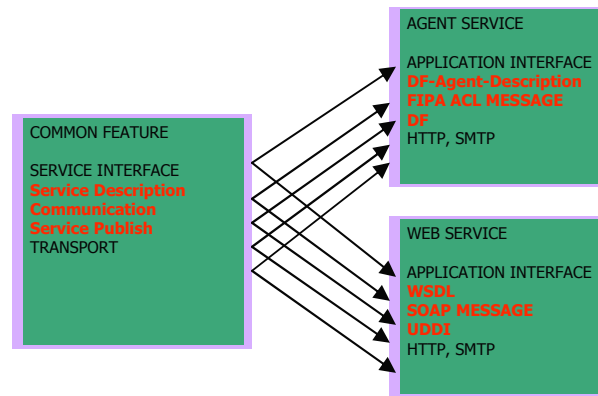


Figure 1: Architectural Aspects of Agent service and Web Service

So again most important issue noticeable from here is that, Agent based System and Web Services both are Software Applications and successful implementation is possible using current technologies (XML, JAVA).

In architectural perspective, web services and agent services both services have some common features which are service interface, service description, communication, service publish and transport [figure 1]. The differences found from the above features are in service description, communication and service publish which are in their specifications. For these tree above mentioned features agent services follow FIPA specifications namely DF-Agent-Description for service description, DF for service publish in multi-agent platform, ACL for agent services communication. On the otherhand web services follow W3C specifications which are WSDL for service description, SOAP for communication and UDDI for service publish. For transportation both technology uses the same protocol(HTTP, SMTP).

The main differences to provide interoperability between web services and agent services are inside specifications. Specification for publishing a service in UDDI and DF-facilitator are quite different. They based on different syntax and semantics and service policy, level of security are also different. So to come up under a common frame work it is mandatory to change in specification for both technologies and another possibility

would be orientation of a new of distributed middleware for inter-operability of these two emerging technologies to come under a common framework. From recent survey of web services and agent services, we found that the mostly used framework for multi-agent systems is JADE framework, which doesn't provide sound mechanism to bind web services or registry a web service in a agent service repository so call directory facilitator. Now many researcher and developer claim that, AXIS2 [2] is the next generation web service middleware with the feature of security, reliable messaging, co-ordination addressing also with support for REST. But the main problem they are not providing any bridge or middleware to integrate web services and agent services. So what is the possible solution to work together with agent and web services.

3. Ideas to Solve Inter-operability issue of Web Services and Multi-Agent Systems

According to our analysis the problem is not so easy. So we are not considering the problem a trivial one or a programming task, as our goal is to finding an optimal solution which can fit with these two technologies in a way that both can be facilitated from the features of each other. So lots of effort needs to provide an optimal solution regarding this problem. As web service developer point of view, usually the developer wants the build in library functions or the easiest options to access an agent service. Developer needs not to care about the architectural issues behind the development of a multi-agent systems framework. On the other hand Multi-Agent systems developer doesn't care about the architectural issues of web services, his requirement is to get the easiest way that agent services can easily communicate with the web services. Our finding regarding the solution of this demanding issue is to provide a middleware which works as gateway for both technology and another solution could be the change in specification of communication ACL and SOAP specifications.

Web Services Middleware to invoke and operate agent services: Typically web services need to perform three kinds of operations use agent services namely finding service, publishing service and binding service. So the task of Web Service middleware is to provide the mechanisms that web services can perform the above operations. From the current standard middleware of web services, the above mentioned operations are only applicable to bind web services or business processes. AXIS2 middleware [2] which supports SOAP messaging for Web Services communication extract from W3C specification. It also supports widely popular REST style Web Services. The specifications supported from AXIS2 are SOAP 1.1 and 1.2 message transmission optimization technique (MTOM), XML optimization Packaging (XOP), SOAP with attachment, WSDL 1.1, WS-Addressing, WS-Policy SAAJ 1.1. Our idea is to add also ACL specification given by FIPA (Federation of Intelligent Physical Agents) for Multi-agent communication with the objective to provide accessibility to publish, find and bind agent services in web services environment also multi-agent environment.

Multi-Agent System Middleware to invoke and operate Web services: In the same way we can extend JADE (Java Agent Development Framework) [1] to support web services to communicate with JADE based multi-agent systems. In this case, the proposal is to support SOAP 1.1, SOAP 1.2 WSDL 1.1 WS-Addressing and WS-Policy.

The third solution is to develop middleware applications which will understand both W3C and FIPA specifications [5]. The concept is to develop two applications in which one would be gateway service and the other would be gateway agent. When a web

service needs to interact with agent service, it will forward its request to gateway service. This gateway service is built on W3C specifications and task is to communicate with FIPA based agent services to perform the operations initiated by a web service. On the other hand when a agent service needs to interact with web services, the agent service forward the request to gateway agent. Gateway agent is portable with FIPA specifications and its objective is to receive request messages from agent service and process this message as W3C portable messages and forward it to desired web services. And after getting response forward the response message to the agent services in FIPA specifications.

4. Implementation

We haven't started implementation yet but different kind of simulation is going on to support the ideas. We have developed some small services to support simulations. Soon we will start implementation.

5. Conclusions

In this paper, we have tried to come out with concrete ideas to change the way of providing Inter-operability for the current web. MAS and Business processes are becoming popular for their tremendous advance features for Business Automation. We believe that if we able to provide a sound middleware which can completely take away the distance between web services and agent services, it will be a great contribution in current e-commerce and this is the future goal of this work.

Reference

1. JAVA Agent Development Framework JADE <http://jade.cselt.it/>
2. Next Generation Web Services Middleware AXIS2 <http://ws.apache.org/axis2/>
3. Federation for Intelligent Physical Agents FIPA <http://www.fipa.org/>
4. Web Services Activity W3C <http://www.w3.org/2002/ws/>
5. M. Omair Shafiq, Ying Ding, Dieter Fensel, "Bridging Multi Agent Systems and Web Services: towards interoperability between Software Agents and Semantic Web Services," *edoc*, pp. 85-96, 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06), 2006
6. M. Wood and S. A. DeLoach. An overview of the multiagent systems engineering methodology, 2000.