Service Interoperability between Agents and Semantic Web Services for Nomadic Environment

Adeel Shajjar¹, Naeem Khalid¹, Hafiz Farooq Ahmad¹,², Hiroki Suguri²

¹NUST School of Electrical Engineering and Computer Science
166-A, Street 9, Chaklala Scheme 3,
Rawalpindi, Pakistan
{adeel.shajjar,naeem.khalid, drfarooq}@niit.edu.pk
²Communication Technologies, 2-15-28 Omachi Aoba-ku, Sendai 980-0804 Japan
{suguri, farooq}@comtec.co.jp

Abstract

In current information age, lives of people are driven by the availability and utilization of services to get information any time, any where, in any format, on any device. For a nomadic user, information becomes useful if certain scales of autonomy and intelligence are present in the information systems. In this context Lightweight Multi Agent System (L-MAS) becomes a preferable choice for design and development of intelligent autonomous mobile information systems. In this paper we aim to propose a multi agent based system architecture to bring OWL based semantic web services to nomadic users. We propose a design for smart nomadic client using, L-MAS, which interacts with a multi agent based mediator system to get access to the OWL based semantic web services.

Keywords: Multi Agent Systems (MAS), smart nomadic client, web services, semantics, interoperability, mobile information systems.

1. Introduction

The semantic web [1] aims at incorporating semantic capabilities to the information contents available over the World Wide Web. Web services provide standard programmatic interfaces for interoperability between different software applications over the World Wide Web [2]. Web services are enhanced to semantic web services by W3C Semantic web service collation group [3]. In order to provide semantic representation of web services, ontologies [14] are used. W3C has proposed two ontology languages for representing knowledge on semantic web: 1) Resource Description Framework Schema (RDFS), based on XML and logic programming, 2) Ontology Web Language (OWL) [9] based upon description logic and it provides constructs for cardinality restrictions, Boolean expressions and restriction on properties [14].

On other hand Multi Agent Systems (MAS) [4] provide an application environment where several sophisticated software entities (Agents) work in autonomous and intelligently fashion, collaborating with each other to achieve certain goals on behalf of users or other system components.

Multi Agent Systems and Web Services have eminent communication hurdles due to the differences in the technological specifications [5,6], governed by two autonomous bodies, Foundation for Intelligent Physical Agents (FIPA) [4] and World Wide Web Consortium (W3C) [3] respectively. We have already proposed Ontology Gateway [12] and Middleware Ontology Services [14] for facilitating interoperability between MAS and semantic web services.

Albeit the technological advancements, the resource capabilities of handheld devices in terms of memory and processing power yet cannot compare with the resource capabilities of personal computers [10]. These resource constraints limit the scale of functionality that lightweight Multi Agent Systems (L-MAS) could provide to process and disseminate useful information in timely manner. Semantic web service provisioning to handheld devices using L-MAS becomes even more challenging due to resource limitations of handheld devices.

In order to provide web services to nomadic users, three architectural options are reported [7]: wireless portal network, wireless extended internet, and wireless ad-hoc network (peer-to-peer). A wireless portal network uses a mediator between a mobile client and web service providers; this mediator is responsible for the conversion of XML messages to device supported message format. Wireless extended internet enables mobile clients to access web services directly. Wireless ad-hoc network enables the mobile clients to act as service provider and service consumer in a peer-to-peer environment.
In this paper we aim to propose a mediator based architecture, using multi agent technology, to bring OWL [9] based semantic web services to nomadic users.

The remainder of this paper is organized as follows: Section 2 provides a brief description of interoperability issues between L-MAS, MAS, and semantic web services. In Section 3, we present our proposed architecture. Section 4 explains the detailed design of the proposed architecture. In Section 5 we present evaluation and future work. In Section 6 we sum up the paper providing conclusions of this research effort.

2. Interoperability Issues

In this section we present a brief description of current interoperability issues under the shadow of available scenarios for semantic web services provisioning to handheld devices using Multi Agent Systems.

2.1. MAS and semantic web services interoperability issues

MAS and web services have different technological specifications. The differences arise in the following areas [5,6,12,14].

**Differences in communication protocols:** Agents use Agent Communication Language (ACL) and Web Services use Simple Object Access Protocol (SOAP).

**Differences in service description languages:** Agents use ontology named Directory Facilitator Agent Description (DF-Agent-Description) whereas Web Services use Web Services Description Languages (WSDL).

**Differences in service registries:** Agents have Directory Facilitator (DF) based on FIPA specifications, whereas Web Services use Universal Description Discovery and Integration (UDDI) which is based on W3C specifications.

**Differences in Ontology descriptions:** FIPA-SL [FIPA00008] is FIPA recommended language to develop ontologies for agents and OWL [9] is a W3C recommended language to develop ontologies for semantic web. As these languages are based on two different standards so mapping between these two languages is a challenging task and is quite demanding.

Several middleware based approaches have been developed to overcome these differences, to enable interoperability between MAS and Web Services [5,6]. The core objective of such kind of a middleware is to provide effective and efficient bidirectional translation between, FIPA-SL and OWL, ACL and SOAP, to facilitate service description and service utilization respectively.

We have already designed and developed a Middleware Ontology Services [14] in order to tackle the issues regarding interoperability of MAS and semantic web services.

2.2. L-MAS and Semantic Web Service Interoperability Issues

L-MAS and semantic web service interoperability, inherits the same issues as discussed in subsection 2.1. In this scenario situation becomes even more complex as the functionality of L-MAS is restricted due to the constraints imposed by the programming languages and capabilities of handheld devices. Web services require interaction in terms of SOAP, represented entirely in XML. Implementation of an XML based direct solution, in a nomadic environment, lead to an inefficient approach because they introduce extra processing and resource overhead for already resource limited devices [15,16]. The overhead incurred in XML message representation as compared to HTTP is reported to be around 400% [15].

2.3. MAS and Light-weight MAS Interoperability Issues

FIPA ACL defines the syntax, semantics, and pragmatics for the agents to communicate with each other [FIPA00061]. FIPA has defined a bit efficient encoding scheme specification for ACL messages [FIPA00069] and message envelopes [FIPA00088]. This scheme is based on EBNF and is suitable for wireless networks due to compact nature of messages, which need to be transferred, over a low bandwidth wireless link [10] e.g. GPRS. MAS developed for wired networks often do not use bit efficient encoding scheme for ACL messages and envelopes. To enable interoperability between such a MAS and L-MAS, a conversion mechanism is required to convert ACL messages and envelopes to bit efficient encoded ACL messages and envelopes, and vice versa. Moreover language specification for an agent must be set to FIPA-SL0, which is a minimal subset of FIPA-SL, in order to reduce communication overhead for mobile client. Management of frequent connection and disconnection of a mobile client needs to be monitored gracefully, in order to provide accurate information at right time.

3. Proposed Architecture

In this section we describe our proposed architecture, which facilitates interoperability between L-MAS, MAS and OWL based web services, using middleware based approach. We have addressed the issues mentioned in section 2
regarding nomadic environment, by designing our architecture in accordance with FIPA nomadic application support specification [FIPA00014].

Message representation and transmission issue in our proposed architecture is tackled through multi agent based smart nomadic client. Multi agent based smart nomadic client is designed using L-MAS and it receives information, as much as it can easily process on a resource constrained device, such as it handles user preferences, user mobility information, connection management, interactive user interface, and quality of service (QoS) issues [FIPA00094].

MAS mediator provides all the intensive XML/SOAP based information processing. Figure 1 delineates the essential components of the Proposed System architecture.

3.1. Participant Agents

FIPA nomadic application support specification provides the details of two types of agents: Monitoring Agent (MA), and Control Agent (CA). Other then these agents there are application agents. Our proposed architecture introduces UA, SBA, and OA as application agents, which are responsible for the semantic based web service request, execution, and response. A brief description of these agents and their roles is as follows.

3.1.1. Monitoring Agent (MA). MA consists of network service specific components and is responsible for collecting, storing and providing analysis of raw performance data (e.g. available device memory, strength of communication channel signals, etc). In our proposed architecture, MAiM and MAiF represent monitoring agents in L-MAS and MAS mediator respectively.

3.1.2. Control Agents (CA). CA is responsible for the management of Message Transport Service (MTS). CA selects appropriate Message Transport Protocol (MTP), Message Transport Channel (MTC), and manages the transmission and reception of messages. In our proposed architecture, CAiM and CAiF represent control agents in L-MAS and MAS mediator respectively. CAiF resides on communication middleware within MAS mediator. CAiF uses FIPA Bit-efficient-ACL codecs and FIPA Bit-efficient-Envelop codecs to provide interoperability between L-MAS and MAS mediator.

3.1.3. User Agent (UA). UA resides on mobile agent platform and is responsible for providing a user interface to the nomadic user. User submits the service query and view service response using this interface. UA captures user’s preferences regarding a service and reformulates the query by getting device related information from MAiM. UA manages the collection and storage of user preferences for future service requests.

3.1.4 Service Broker Agent (SBA). SBA is responsible for searching and querying ontologies in FIPA-ACL format. SBA maintains a list of registered OA and maintains domain specific information about the ontologies and ontology services they offer.

3.1.5. Ontology Agent (OA). OA resides on Ontology Services Middleware and is responsible for ontology specific tasks. Its main responsibilities include translation, storage, and management of ontologies.
4. Detailed Design

In order to realize a communication system, mutual understanding of message format is a prerequisite for entities interested in communication. In this section we present the details of middlewares used in our proposed architecture.

4.1. Communication Middleware

In our proposed architecture we are using General Purpose Radio System (GPRS) as communication channel and Hyper Text Transfer Protocol (HTTP) as communication protocol. Upon startup of MAS mediator system, all agents register themselves and their service capabilities with DF. When CAiF registers itself, it initializes HTTP Mobility Aware Server component which continuously listen for mobile client requests. Steps involved in communication between L-MAS and MAS mediator are as follows:

**Step 1:** When a UA is created or activated on L-MAS in pursuit of a semantic web service execution it needs to establish, or use already established, communication channel with MAS mediator. For establishment of communication channel UA requests CAiM of L-MAS to propose a communication commitment to CAiF of MAS mediator. CAiM contains a list of known MAS mediator access points, set as standard access points. CAiF requests Agent Management System (AMS) for grant and registry of communication link.

**Step 2:** Upon receiving permissions from AMS, CAiF notifies CAiM about the acceptance of communication proposal.

**Step 3:** After receiving acceptance notification, CAiM establishes a connection with MAS mediator.

**Step 4:** Using established channel UA registers itself with the DF of MAS mediator to publish its services to agents residing on mediator system. This registration is temporal and depends upon the connectivity of the communication link.

4.2. Ontology Services Middleware

Ontology Services Middleware is responsible for interoperability between MAS and semantic web services. Upon startup of MAS mediator, OA registers itself with DF along with translated ontologies. OA is responsible for translation of message format and content language according to the nature of target recipient. FIPA-ACL-to-SOAP and FIPA-SL-to-OWL components are used to translate message format and content language respectively, if recipient is a Semantic Web Service System. SOAP-to-FIPA-ACL and OWL-to-FIPA-SL components are used to translate message format and content language respectively, if recipient is an agent. Whenever an agent, from MAS, wants to interact with a semantic web service, OA provides the translations and forwards the query to UDDI-OWL-Matchmaker. Matchmaker returns the address, Universal Resource Identifier (URI), of the required service to OA. OA then extracts the web service from target address. After extracting the target service, the service ontology is passed to OWL-to-FIPA-SL component for identifying ontology concept schema, predicate schema, aggregate schema, and agent-action schema, in order to formulate equivalent FIPA-SL ontology.

OA uses Ontology Web Server to save translated ontologies after their registration with SBA. Registration with SBA makes the translated ontologies visible to agents in MAS. So next time when the same service is requested OA already knows the address of the service and it no more requires extra interaction with UDDI-OWL-Matchmaker to retrieve service address of already registered service. OA manages the versioning and alignment of registered ontologies using Ontology Management component. Using Ontology Management Component, OA keeps the track of correct service addresses and to accommodate the knowledge of changed address of same service.

5. Implementation

In this section we present the details of our proposed architecture from implementation perspective.

5.1 Communication Channel Establishment: Figure 2 shows a communication sequence diagram to
establish an HTTP based communication channel between a mobile client and mediator system. A UA from mobile device issues a request to the CAiM to activate the fipa.mts.mtp.http.std MTP. ACL Messages for communication channel establishment are as follows:

**Message 1: Connection establishment request.**

```xml
<Message>
  <request>
    <sender>
      <agent-identifier>UA@sagelite.com</agent-identifier>
    </sender>
    <receiver>
      <agent-identifier>CAiM@sagelite.com</agent-identifier>
    </receiver>
    <ontology>fipa-nas</ontology>
    <language>fipa-sl</language>
    <protocol>fipa-request</protocol>
    <content>
      (action
        (agent-identifier>CAiM@sagelite.com</agent-identifier>
        (activate (sequence
          (transport-protocol name=fipa.mts.mtp.http.std
          :dest-addr http://niit.edu.pk:8556/acc)))))
    </content>
  </message>
</Message>
```

**Message 3: Connection proposal for CAiF**

```xml
POST http://niit.edu.pk:8556/acc
HTTP/1.1
Cache-Control: no-cache
Host: niit.edu.pk:8556

<?xml version="1.0" encoding="UTF-8"?>
<propose>
  <sender>
    <agent-identifier>CAiM@sagelite.com</agent-identifier>
    <receiver>
      <agent-identifier>CAiF@niit.edu.pk</agent-identifier>
    </receiver>
    <ontology>fipa-nas</ontology>
    <language>fipa-sl</language>
    <protocol>fipa-propose</protocol>
    <content>
      (action
        (agent-identifier>CAiM@sagelite.com</agent-identifier>
        (use
          (transports
            :send (sequence
              (transport-protocol name=fipa.mts.mtp.wap.std)
            )))
          (recv (sequence
            (transport-protocol name=fipa.mts.mtp.http.std)
          )))
        true)
    </content>
  </propose>
</Message>
```

5.2 Service Request: Figure 3 shows the sequence diagram for service request. Following ACL message shows the service request from CAiM to CAiF.
Ontology search: Following ACL message shows the service related ontology request from SBA to OA.

Inform:

5.3 Ontology search: Following ACL message shows the service related ontology request from SBA to OA.

6. Evaluation and Future Work

At NUST-COMTEC lab we have developed a test application to validate our proposed architecture. We used SAGE-Lite [10] as L-MAS and SAGE [13] as MAS mediator. We developed a prototype, W3C compliant semantic web, system to access thesis record of MS students. We developed OWL based ontologies to describe the contents of the research records, to facilitate semantic based searching. MAiF, CAiF, OA, and SBA were created within SAGE environment. UA, CAiM, and MAiM were created using SAGE-Lite. For deployment of the system we used SAGE with Axis web server to host research record web services and used Jena API to create ontologies, for record descriptions, in OWL. To deploy a smart nomadic client we used J2ME wireless toolkit 2.5 and its default color phone emulator. We received successful communication results between SAGE-Lite and SAGE. The translations between FIPA-SDL and OWL ontologies and FIPA-ACL and SOAP messages were successfully accomplished autonomously.

Our future work includes deploying the smart nomadic client on a smart phone to get real time data for performance analysis. In near future our intention is the design, development, and deployment of an intelligent Student Management System, to facilitate the students of NUST and other universities, to coordinate research related activities using our proposed architecture.

7. Conclusion

In this paper we have presented our approach for providing Semantic Web Service access to handheld device clients, using L-MAS and MAS. We tried to exploit the advantages of mediator based middleware approach. This scheme suits the communication requirements of handheld devices over a wireless communication link using HTTP. The proposed architecture provides a good foundation for the development of integration systems for semantic web based systems.

8. References


