System Architecture for Interoperability between OWL and FIPA ACL

By

Maruf Pasha
(2004-NUST-MSIT-27)

NUST Institute of Information Technology
National University of Sciences & Technology
Rawalpindi, Pakistan
May 2006

System Architecture for Interoperability between OWL and FIPA ACL

By

Maruf Pasha
(2004-NUST-MSIT-27)

Supervisor

Dr. Hafiz Farooq Ahmad

A thesis submitted for the fulfillment of the requirements for the degree of

Masters in Information Technology

In

NUST Institute of Information Technology
National University of Sciences and Technology
Rawalpindi, Pakistan

(May 2006)

CERTIFICATE

It is certified that the contents and form of thesis entitled "System Architecture for Interoperability between OWL and FIPA ACL" submitted by Maruf Pasha have been found satisfactory for the requirement for the MS (IT) degree.

Advisor: ______________________

Associate Professor (Dr. Hafiz Farooq Ahmad)

Co-Advisor: ______________________

Director General (Prof. Dr. Arshad Ali)

Committee Member 1: ______________________

Lecturer (Mr. Muhammad Atif)

Committee Member 2: ______________________

Dr. Hiroki Suguri

DEDICATION

I would like to dedicate my work to my parents, whose sincere prayers and love are a constant source of strength for me. They have always encouraged me towards pursuing higher studies and back me up for whatever decision I made.

4th July 2006

Maruf Pasha
(2004 – NUST-MSIT-27)

The report was examined in the light of the HQ NUST format. There were some minor mistakes (e.g. list of tables wasn't there and 1 or 2 page numbering wasn't correct etc). The mistakes were briefed to Mr. Maruf Pasha accordingly and he has resubmitted the report today after making the appropriate corrections.

Forwarded for further action.

Thesis Scrutinizing Committee:

Wg Crd (R) M. Ramzan

4/7

Mr. Danish Hassan Khan

CERTIFICATE OF ORIGINALITY

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NIIT or any other education institute, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at NIIT or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic is acknowledged.

Signature: Maruf

Maruf Pasha

ACKNOWLEDGMENTS

First of all I am thankful to Allah Almighty for giving me courage and spirit to complete this highly challenging task and to compete with international research community. I am also grateful to my family, especially my parents who have supported and encouraged through their devotion towards me and my studies and their prayers are always there with me.

I would like to express my most sincere gratitude to my advisor, Dr. Hafiz Farooq Ahmad who helped me walk through such rigorous but precious period in my life. Under his guidance and with his suggestions I clarified the directions and overcome the difficulties occurred during the development of my thesis. His strict thinking, hard working, and attitude of stepping on solid ground inferences my way of thinking, doing things and facing challenges. It was my greatest honor doing research with him.

I am highly thankful for the prompt guidance and help I received from Prof. Dr. Arshad Ali, Dr. Hiroki Suguri.I am very much thankful to Mr Muhammad Atif for his valuable suggestions throughout my research. I am thankful to Prof. Dr. Arshad Ali for his continuous and valuable suggestions and guidance, especially for the provision of all kinds of facilities throughout this research work. His ability of management and foresightedness taught me a lot of things which will be more helpful for me in practical life.

I am highly thankful to all of my teachers whom had been guiding me through out my course work and increased my knowledge. Their knowledge, guidance and training helped me a lot to carry out this research work.I would like to offer my appreciation to all the project team and my close colleagues, who had been encouraging me throughout my research work, Muhammad Bilal, Sabih ur Rehman, Sarmad Sadik, Muzammil Ahmad Khan , Mujtaba Haider, Sohail Awan ,Kazim Ali.

Maruf Pasha

Table of Contents

Chapter 1 13

INTRODUCTION 13

1.1Scope of Research Area 16
1.2 Rational for Research 18
1.3 Problem statement 19

Chapter 2 20

LITERATURE REVIEW 20

2.1. ONTOLOGIES 20
2.2. Role of Ontologies 21
2.3.Sample Ontology 22
2.4 Personalizing WebServices through Agents 22
2.5 Automated Ontology Generation 23
2.6 Agents in Science Projects 23

Chapter 3 25

MULTI AGENT SYSTEMS 25

31 AGENT 25
3.2 AGENT PLATFORM (AP) 26
3.3 MULTI-AGENT SYSTEMS (MAS) 26
3.4 FIPA (Foundation for Intelligent Physical Agents) 27
3.5 Agent Management System (AMS) 28
3.6 Message Transport Service (MTS) 29
3.7 Agent Communication Language (ACL) 29
3.8 FIPA-SL 30
3.8.1 FIPA-SL0 31
3.8.2 FIPA-SL1 31
3.8.3 FIPA-SL2 31
3.9Directory Facilitator 31
3.10 Visual Management Agent (VMA) 32

Chapter 4 34

GRID COMPUTING 34

4.1 Grid Services 35
4.2 Semantic Grid 36
4.3 Semantics for Grid Computing 37
4.3.1 Semantic Grid services 37
4.3.2 Integration of Information 38

Chapter 5 40

COMPARATIVE ANALYSIS OF FIPA-SL AND OWL 40

5.1 Expressiveness of OWL and FIPA SL 42
5.2. Representing Agents Attitude 44
5.3. Comparison based on Logic 45
5.4 Issues between interoperability of OWL and SL 46

Chapter 6 48

PROPOSED ARCHITECTURE 48

6.1 Software Agents communication with the OWL based Web Services 50
6.2 Grid entities communication with the Software Agents 52

Chapter 7 54

CONCLUSIONS AND FUTURE DIRECTIONS 54

7.1 CONCLUSION 54
7.2. FUTURE DIRECTIONS 55

Chapter8 56

RESULTS AND DISSUSIONS 56

8.1. OWL to SL Ontology Conversion 56
8.2. SL to OWL Ontology Conversion 56
8.3. Ontology Conversion Comparison 57

REFERENCES 58

APPENDIX 59

Test Bed Specifications 59

LIST OF ABBREVIATIONS

MAS Multi Agent Systems

SAGE Scalable fault tolerant Agent Grooming Environment

FIPA Foundation for Intelligent Physical Agents

AI Artificial Intelligence

AMS Agent Management System

MTS Message Transport Service

ACC Agent Communication Channel

MTP Message Transport Protocol

DF Directory Facilitator

ACL Agent Communication Language (ACL)

AID Agent Identifiers

SL0 Semantic Language 0

SL1 Semantic Language 1

SL2 Semantic Language 2

OWL Ontology Web Language

XML Extensible Markup Language

HTTP Hyper Text Transfer Protocol

SOAP Simple Object Access Protocol

WSDL Web Services Description Language

UDDI Universal Description and Discovery Integration

RDF Resource Description Framework

OWL-S Ontology Web Language for Services

API Application Programming Interface

DAML	DARPA Agent Markup Language
DAML-S	DARPA Agent Markup Language for Services
OIL	Ontology Inference Layer

List of Tables

Table 1: Comparison of OWL and FIPA Ontologies —— 41

List of Figures

Figure 1: Essentials of Agent Communication Language —— 14

Figure 2: Scope of Research Area —— 18

Figure 3: Integration of Multiple Technologies —— 19

Figure 4: University Ontology —— 22

Figure 5: Components of FIPA compliant MultiAgent Systems —— 28

Figure 6: Proposed System Architecture —— 49

Figure 7: Software Agents communicating with the OWL based Services —— 51

Figure 8: Grid entities communication with Software Agents —— 52

Figure 9: OWL to SL Conversion —— 56

Figure 10: SL to OWL Conversion —— 56

Figure 11: Ontology Conversion Comparison —— 57

ABSTRACT

Evolution of World Wide Web has made it a focal point of research to apply structured and well-defined meanings to achieve Semantic Web and ultimately the Semantic Grid. The realization of this goal gives rise to the integration of technologies that can negotiate and cooperate in which agents have proven to be effective. OWL (Web Ontology Language) is currently the W3C standard for providing the explicit semantics to the web services whereas the FIPA Semantic Language is the core of agent platforms due to its high expressive power. The key objective of this paper is the development of semantic translations in such a way that the agents can communicate with the web services in an efficient manner. The challenging domain where our work can be applied is the messages that can be exchanged between FIPA compliant Multi Agent systems and the web services, thus enabling the agents based applications to interact with the web services. Our goal for these mappings/translations is to show that how the existing standards of FIPA Semantic Language could be mapped into representations, which could be readily used in a Web Services, based environment.

This dissertation describes that interoperability is required for effective means of communication in multi-agent systems. The communication architecture is proposed for the agents and web services using the idea of an Ontology Gateway. The issues and trade-offs between OWL (Ontology Web Language) and FIPA-SL (Semantic Language) is analyzed when used as semantic language for communication between agents. OWL is less expressive as it is based on Description Logic whereas the FIPA Semantic Language is based on First Order Logic.

Chapter 1

INTRODUCTION

The concept of the Semantic Web has provided the foundations for the autonomous interoperation between the entities on the Internet. This interoperability can be achieved through annotation of the contents at arbitrary locations on the Web with machine processable data. When such annotations are associated with the ontologies, then computers can attain a certain degree of understanding of the data. Ontologies [4] are formal and explicit specifications of certain areas and are collective between large groups of stakeholders and are core of the Semantic Web. These properties make ontologies ideal for machine processing and facilitating interoperation and thus enhancing the interoperability and machine understandability. In fact, ontologies form the core of Semantic Web and are the key to enable automated interoperation and cooperation. Ontology is a combination of classes, their relation or properties, instances and axioms, expressed in some formal language.

Currently the computers can just read the contents and cannot understand them well i.e. machines can parse the web contents for layout and processing for example navigational links to some other pages etc. The evolution of semantic web will give rise to structure the meaningful contents of WebPages and it will be an extension of the existing web in which the information processed is given a well-defined meaning and better-enabled computers [1]. To efficiently discover and utilize the resources on the web, the semantics would act as epoxy resin for autonomy. To bracket together the explicit semantic representations in the web services several ontologies and ontological web languages have been anticipated, which not only have deficiencies but also impose an overhead. W3C and other research groups are doing a lot of efforts to standardize these languages in order to integrate the semantics in existing languages. In order to allow sharing and reuse of ontologies on the Semantic Web, a common ontology language is required. The W3C has proposed two ontology languages for use on the Semantic Web. The first is RDFS [6], based on XML [9] and logic programming, which is a lightweight ontology language. Whereas OWL [7] is another language by W3C with more expressive ontology language based on Description Logics [5][2].

On the other hand (MAS) Multi Agent System is a promising field of distributed artificial intelligence, where an agent is a computer system capable of flexible autonomous action in a dynamic unpredictable and open environment [8]. The exploitation of software agents in daily life application is increasing day by day and is becoming essential roles in the heterogeneous environments. The agent properties that make them popular among different domains are autonomy, intelligence, mobility and communication.

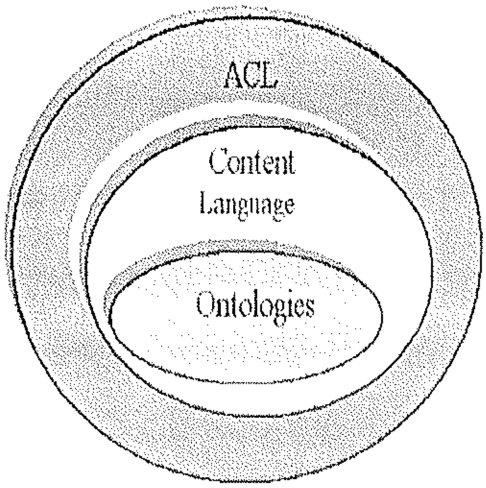

Fig. 1. Essentials of Agent Communication Language

To hold a meaningful conversation between humans or agents communication languages play a vital role with explicitly associated semantics for an effective communication. In general communication is the intentional exchange of information brought about by the production and perception of signs drawn from a shared system of conventional signs. The property of an agent to communicate and interact with other agent is the most important aspect in MAS. Agent communication is referred as interaction in which the dynamic relationship between agents is expressed through the intermediary of signals that once interpreted, will affect these agents [3]. Agents require a formal language to carryout effective and meaningful conversation as most of them are engaged in employing some knowledge on behalf of user objective and desires.

Expressive power of agents and semantics is one of the issues that affect successful communication between multi-agent systems. Lack of expressive power in agents restricts them to communicate and understand high-level messages between them. Also the message should be declarative, simple and readable by people so that agents understand them and they can be easily extended for further enhancements in the vocabulary. The messages should be easy to parse and generate because the content is encoded using a declarative knowledge representation language called the semantic or content language proposed by FIPA. The semantics should be unambiguous for agents to understand the messages fully.

Content languages are way of expressing contents that are being transferred between agent or web services. A good content language should be able to express rich forms of content and can be efficiently processed and fit well with existing technology. The content language is a language used to express the content of messages exchanged between agents. It allows expressing an action, its internal results and its arguments. The agent itself determines how to extract the action and its attributes to carry out the invocation. The content language describes how a message is encoded on the wire while in transfer between two agents. A number of different languages have been proposed like FIPA-SL, XML, RDF and now OWL (Ontology Web Language) which is based over XML and RDF, and is a new candidate to be used as semantic language in MAS.

To Engineer such a system with all the above mention problems was a very challenging task. As all the interactions among agents takes place through ACL Messages so the semantic language in ACL message structure should have Rich Semantics and higher order of Expressivities to resolve the above mentioned problems. Similarly the protocol used for the communication between web services is SOAP and the ontologies are described in Web Ontology Language. To improve the communication of agents and Web services the Semantic Language should be interoperable with the Semantic Language. So the communication issues as well as interoperability issues are dealt in my dissertation.

1.1 Scope of Research Area

This project falls in the domain of Autonomous Semantic Grid which has three major emerging technologies like Web Services Framework, Grid Computing and Multi Agent Systems. Web Service Framework is based on ideology of service oriented computing for providing loosely coupled, implementation neutral and heterogeneous resources. Grid Computing focuses on coordinated resource sharing among dynamic virtual organizations.

The Semantic Web is an idea of WWW inventor Tim Berners-Lee that the Web as a whole can be made more intelligent and maybe even insightful about how to serve a user's needs. Berners-Lee observes that although search engines index much of the Web's content, they have little ability to select the pages that a user really wants or needs. He foresees a number of ways in which developers and authors can use self-descriptions and

other techniques so that context-understanding programs can selectively find what users want. In Semantic Grid, Web Services further combines with Semantic Web technologies to enable Dynamic web service discovery, invocation, composition, interoperation and execution monitoring.

Multi Agent Systems focuses on systems in which intelligent Software Agents interact with each other. The Agents are autonomous entities that can control their own state. Different Agents can share a common goal or they can pursue their own interests. Multi Agent Systems develop communications languages, interaction protocols, and agent architectures.

Autonomous Semantic Grid project is aimed to provide a framework for open distributed systems and is based on synergy of Web Services, Grid Computing and Multi Agent Systems. By synergy we mean that it would combine properties of the three technologies without disturbing existing specifications to enable semantic interoperability of autonomous entities with each other, and semantically rich description of resources in Grid environment for better utilization. Software Agents will be able to discover resources in grid, form dynamic workflow, compose services, negotiate with other services or Agents to fulfill the agenda of goals.

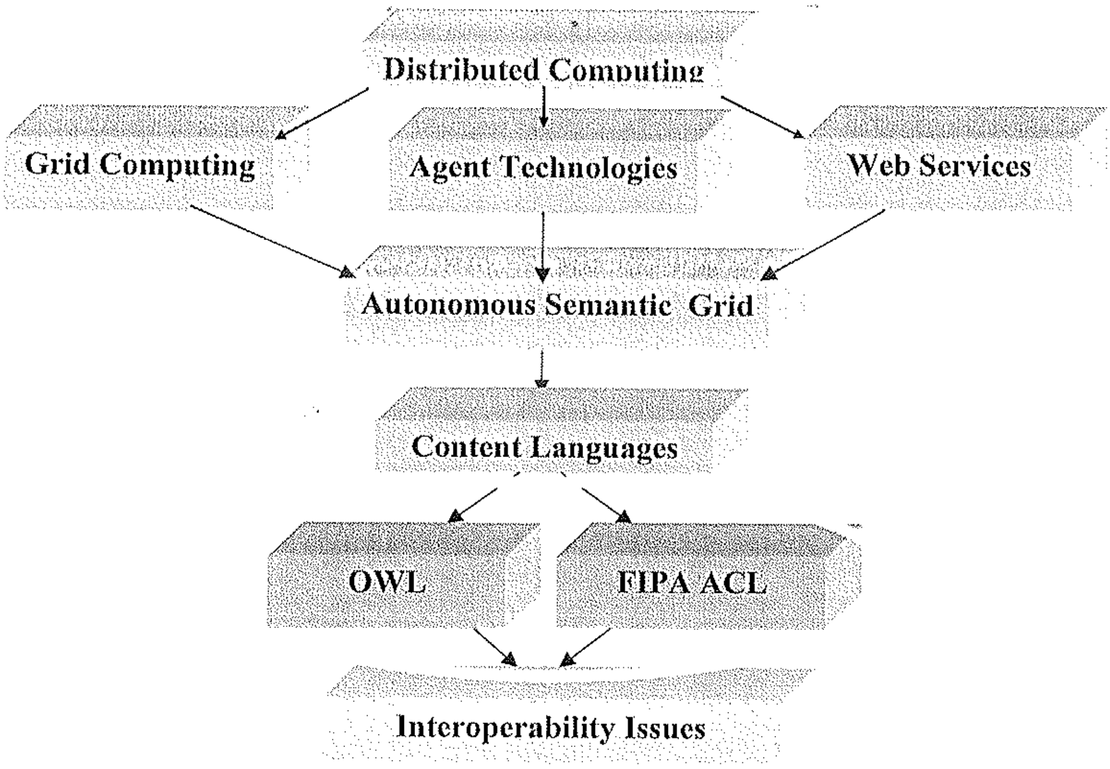

Fig. 2. Scope of Research Area

1.2 Rational for Research

The goal of the Autonomous Semantic Grid is that software agents would be able to autonomously and dynamically able to discover, compose and invoke some particular web services. The information retrieval in the current web is very much difficult as it is based on conventional information processing in which just the text of the documents is processed and there are no semantics. Software Agents and Multi Agent Systems specifications are governed by FIPA (Foundation of Intelligent Physical Agents) and specifications of Web Services are governed by W3C, hence there is a lot of difference among specifications of both technologies and hence Software Agents and Web Service cannot communicate with each other.

1.3 Problem statement

To develop a system architecture for the Interoperability between OWL and FIPA ACL and the integration of agents with the web services without changing existing specification of both the governing bodies. This transformation will facilitate the autonomous agents to dynamically discover and translate the web services. To design and develop an Ontology Gateway as a solution that should act as middleware between Multi Agent System and Web Services communication and without modification of existing standards.

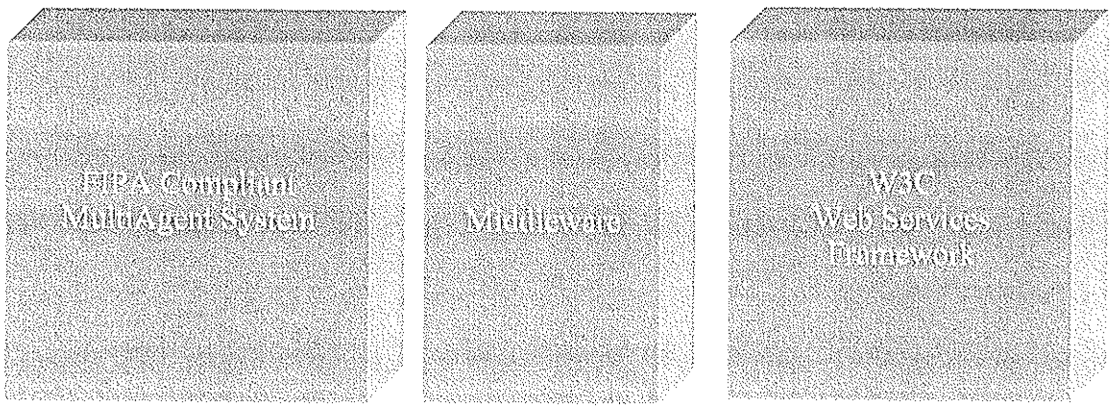

Fig. 3. Integration of Multiple Technologies

Chapter 2

LITERATURE REVIEW

This chapter discusses the related work that has been carried out in this area regarding integration of Software Agents with the Web Services and the role of ontologies.

2.1. ONTOLOGIES

Certain definitions of ontologies are presented by different people. One of the most significant is the next one proposed by Gruber, [Gruber 1993]: ontology is a formal, explicit specification of a shared conceptualization. [Fensel 2001] analyzes this definition identifying four main concepts involved: an abstract model of a phenomenon termed "conceptualization", a precise mathematical description hints the word "formal", the precision of concepts and their relationships clearly defined are expressed by the term "explicit"' and the existence of an agreement between ontology users is hinted by the term "shared" [Fensel 2001]. The definition proposed by Gruber is general; however ontologies can be defined in specific contexts. For example, taking the paradigm of agents into account, [Russell & Norving 1995] establish that ontology is a formal description of the concepts and relations which can exist in a community of agents. The importance of the terms of ontology can be perceived in the next definition: ontology is a hierarchically structured set of terms to describe a domain that can be used as a skeletal foundation for a knowledge base [Swartout et al. 1996]. More recent definitions of ontologies are the following ones:

Ontology is a common, shared and formal description of important concepts in an specific domain [Fensel 2000]. Ontology is a formal explicit representation of concepts in a domain, properties of each concept describes characteristics and attributes of the concept known as slots and constrains on these slots. Sometimes concepts are termed classes, properties are also known as roles while facets are used rather than slots. From these definitions, we can identify some essential aspects of ontologies:

- Ontologies are used to describe a specific domain.
- The terms and relations are clearly defined in that domain.
- There is a mechanism to organize the terms, (commonly a hierarchical structure is used as well as IS-A or HAS-A relationships).

2.2. Role of Ontologies

No communication can take place without a shared, unambiguous and negotiated vocabulary. There are multiple issues while considering vocabulary issues apart from the language barrier. By definition, ontology is a controlled, hierarchical vocabulary for describing a knowledge system [3]. There will be a separate module for sharing a well negotiated and unambiguous ontology.

`An ontology is basically a hierarchical structure that can completely define all the details of any Concept, Action, Predicate and/or a combination of all these. This hierarchy is built by adding schemas into the ontology. Each schema is itself a complete structure of any concept, action, predicate or all of them. A schema can be added into other schemas. There could be schemas of primitive data types as well.

2.3 Sample Ontology

We have used University ontology as to validate our proposed mappings/translations the structure of the sample Ontology is a s follows.

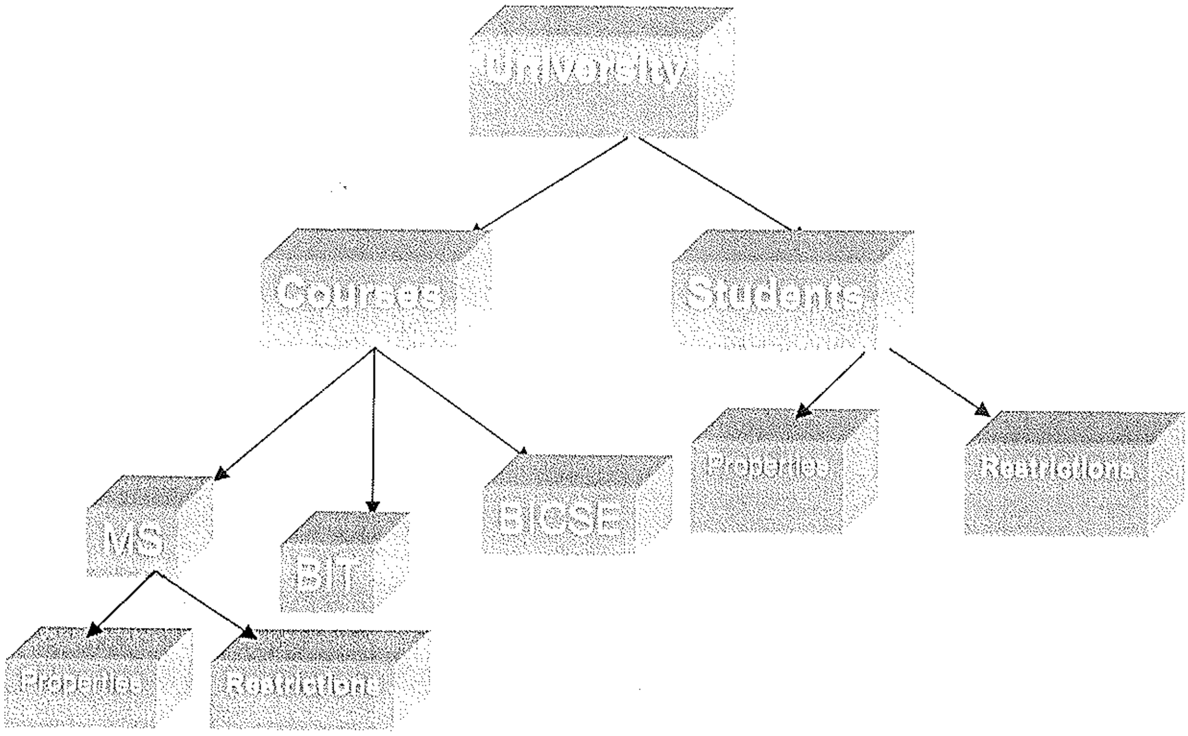

Fig. 4. University Ontology

2.4 Personalizing WebServices through Agents

In [5], the authors answer the commonly raised issue that how the agent technology can be used to personalize web services. They have highlighted a number of critical issues that web service and agent platforms must cater to address in order for the two paradigms to work together, and propose an automated component that can be integrated with existing web service infrastructures. Particularly, they are trying to address the challenge of how a consumer can assign a particular job that will be delegated programmatically and will autonomously be interacting with the web services according to context when acting on the behalf of a consumer.

2.5 Automated Ontology Generation

The ontology that is the automatically generated [15] can be published for the consumer agents that can then send requests to and accept responses from the exemplary agents and performs all the necessary translations between agent communication language messages (agent requests, responses) and SOAP messages (web service calls, results) before and after it calls the web service, respectively.

The solution provided in this paper generates the ontology that describes the web service call signature and the java code of an example agent that can be deployed in any FIPA compliant agent platform.

2.6 Agents in Science Projects

The purpose of this paper [17] was to expose that one can use agent technology to assist in the construction and enactment of e-Science experiments. They have constructed a tool based on this language which allows experiments to be rapidly constructed, verified, and enacted. The language proposed is a lightweight formalism, providing only a minimal set of operations. This was a deliberate choice as it allowed to define the language and the type system without unnecessary complication. Another issue that they intend to address, concerns the discovery of web services. At present, the web services those are used to

define an experiment must be known in advance, and must be explicitly registered before enactment. Furthermore, the protocol must be defined to precisely match the WSDL definition of the web service. In order to reduce the restrictions, and allowing a more extensible kind of coordination, that allows for semi-automatic web service discovery and invocation. For this, they intend to semantically annotate web services, on which one can reason about the behavior of the services.

Chapter 3

MULTI AGENT SYSTEMS

This chapter enlightens some of the important facts about the MultiAgent Systems and the basis for the proposed architecture of SAGE. It also gives an account of the work already being done in this field.

3.1 AGENT

An Agent is an smart software system that has the capability of performing autonomous actions on behalf of its user or owner. Agents in a multi-agent system will be coordinating or acting on behalf of users or owners with very difficult goals or motivations.

The agent is the fundamental cell of the distributed intelligent. Hence, its artificial intelligence (AI) can be built by using all AI implementation technology currently known and it can compute regardless of its current location. This is another property of the agent's technology brilliance; it allows agents to switch their locations by means of migration. In this manner, they are able to do their computing in an environment, which provides more computing power. However, the primary feature of agent technology is the agent's ability to communicate with each other. This enables the agents to unite their efforts to become a collective of working individuals, who are aware of each other's goals and intentions [3].

3.2 AGENT PLATFORM (AP)

Software agents provide multiple services. For the provision of these services, agents require a proper execution environment in which they can execute themselves and keep themselves ready for service provision. Such an execution environment in which agents can be created and can behave according to their specification is called Agent Platform. Many Agent Platforms provide environment for the community of agents for the provision of dynamic services.

3.3 MULTI-AGENT SYSTEMS (MAS)

Multi-agent systems are one of the appealing technologies in software-based framework that provide collaborative environment for a community of social agents for the provision of continuous and dynamic services.

Multiple agents cooperate with one another, normally by exchanging messages through some computer network infrastructure. MAS provide proper execution environment to agents so that they can assure the provision of services to other agents by cooperating, coordinating, and negotiating.

Multi-agent systems (MAS) are relatively new sub-field of distributed systems. Since the beginning of study on MASs in 1980, they have gained a lot of popularity and recognition. The idea of MASs is not tied to a single application domain rather they can be adopted to host different application domains. One of the reasons behind the popularity of MASs is the fact that agents can exploit the opportunities presented by massive distributed systems such as internet.

MAS represent virtual societies where software entities (agents) acting on behalf of their owners or controllers (people or organizations) can meet and interact for various reasons (e.g., exchanging goods, combining services, etc.) and in various ways (e.g., creating virtual organizations, participating to auctions, etc.)

3.4 FIPA (Foundation for Intelligent Physical Agents)

FIPA is one of the standard governing bodies, which provide rules of an abstract architecture for MAS developers to follow. Main objective of FIPA is to develop software standards for heterogeneous and interacting agents and agent-based systems. FIPA focuses on the interpretability issues of MAS. FIPA has standardized a few components within MAS. In order to be called FIPA complaint, implementation of some of those components is mandatory. Figure 1 depicts the architecture of a platform according to FIPA specifications. A typical agent platform contains the following components.

- Agent Management System (AMS)
- Directory Facilitator (DF)
- Message Transport Service (MTS)
- Agent Communication Channel (ACC)
- Message Transport Protocol (MTP)
- Agent Communication Language (ACL)
- Encoding Services

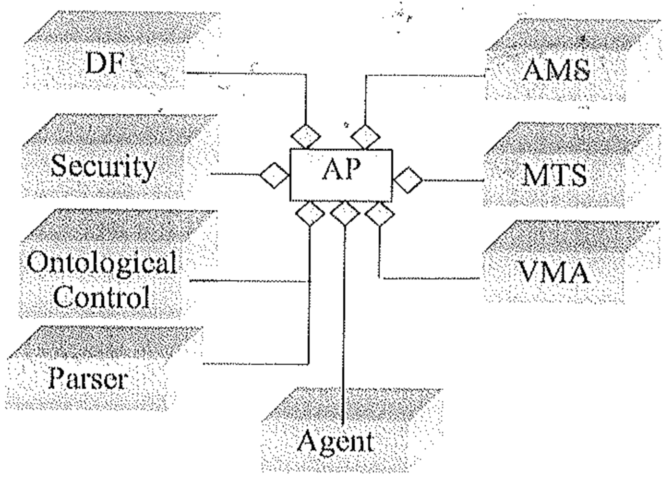

Figure 5: Components of FIPA compliant Agent Framework

3.5 Agent Management System (AMS)

Expected growth of Multi Agent Systems (MAS) with community of social agents in heterogeneous applications has made it focal point for research. All the agents within MAS are managed by Agent Management System (AMS) [10] [11] which is the mandatory supervisory authority of any MAS. A single agent platform can be distributed over several machines which provide scalability and load balancing etc. But with centralized AMS, this infrastructure lacks fault tolerance, which is a key feature of high assurance. Absence of fault tolerance is the main reason for the small number of deployments of MASs. Failure of AMS leads towards abnormal behavior in the distributed platform. Virtual Agent Cluster Paradigm (VAC) is proposed in this regard which strongly supports decentralized distributed AMS to achieve fault tolerance in distributed MAS. VAC is an autonomous distributed infrastructure which provides fault tolerance by using separate communication layers among distributed peers. Experimental results show that it improves performance, brings autonomy and supports fault recovery along with load balancing in distributed MAS.

3.6 Message Transport Service (MTS)

Message Transport Service is the backbone of any MAS. It supports the sending and receiving of ACL messages between agents. The agents involved may be local to a single

Agent Platform or on different Agent Platforms. Two modes of communication are involved for message transportation.

1. Intra-platform Communication (MTS)
2. Intra-machine
3. Inter-machine
4. Inter-platform Communication (ACC)

3.7 Agent Communication Language (ACL)

The ACL package is responsible for creation of a message that's understandable by all entities involved in the multi agent system. Through this package all agents will create a message through some pre defined rules . And the message will be sent to the required

destination. At the reception end, the agent will take its own decision based on the ACL Message.

Agent Communication Languages provides agents with a means of exchanging information and knowledge, which is really the essence of all forms of interaction in multi-agent systems. The result of which, was the FIPA ACL. ACL is an outer language that specifies message format and include descriptions of their pragmatics that is the communicative acts or intentions of the agents. Furthermore, FIPA also define semantic languages to successfully ,communicate with each other. FIPA published SL which provides rich semantics. Every agent has common semantics to talk each other that is based on shared ontology.

3.8 FIPA-SL

The FIPA SL language is a human-readable string encoded (i.e. a content expression in FIPA SL is a string) content language and is probably the mostly diffused content language in the scientific community dealing with intelligent agents. FIPA SL is particularly indicated in open applications where agents from different developers and running in different environments have to communicate.

FIPA Content Languages generally satisfy the some requirements, as SAGE strictly follows the FIPA specification so it also satisfies the following requirements for FIPA-SL

1. They are capable of representing propositions
2. They are capable of representing actions
3. They are capable of representing objects, including identifying expressions to describe objects

To allow simpler agents, or agents performing simple tasks, to do so with minimal computational burden, semantic and syntactic subsets of the full FIPA SL content language for use by the agent are created. These subsets are defined by the use of profiles, that is, statements of restriction over the full expressive power of FIPA SL. These profiles are defined in increasing order of expressivities.

3.8.1 FIPA-SL0

Profile 0 is denoted by the normative constant FIPA-SL0 in the language parameter of an ACL message. Profile 0 of FIPA SL is the minimal subset of the FIPA SL content language. It allows the representation of actions, the determination of the result a term representing a computation, the completion of an action and simple binary propositions.

3.8.2 FIPA-SL1

Profile 1 is denoted by the normative constant FIPA-SL1 in the language parameter of an ACL message. Profile 1 of FIPA SL extends the minimal representational form of FIPA SL0 by adding Boolean connectives to represent propositional expressions

3.8.3 FIPA-SL2

Profile 2 is denoted by the normative constant FIPA_SL2 in the language parameter of an ACL message. Profile 2 of FIPA SL allows first order predicate and modal logic, but is restricted to ensure that it must be decidable.

3.9 Directory Facilitator

Directory Facilitator (DF) is another component of multi agent system. It is accountable for providing yellow-pages directory service to other agents. Agents may register their services to the DF or query the DF to find out what services are offered by other agents. Agent is responsible to provide information related to service i.e servie_type, service_name etc. Furthermore, an agent can also deregister or modify service any agent can interact with a DF in the following situations:

To make its services public, to identify agents that provides a particular service through the yellow-pages FIPA imposes that each Agent Platform has its own DF that is known as default DF. Other DFs may also register with default DF to create a federation.

3.10 Visual Management Agent (VMA)

VMA is an agent that offers a graphical interface to platform administration and platform monitoring. The agent offers many services that show the state of the Agent Platform and it also offers various tools that are used to perform administrative interaction with the AMS agent, the DF agent and are also be used to debug and test applications. The state of the Agent Platform also shows the details of the agents that reside inside the platform. The VMA itself offers some internal agents for platform management and monitoring that can be used to perform different tasks such as:

- Examination of the message exchanges among different agents.
- Create or compose ACL messages and send them to other agents.
- Display the list of all the ACL messages sent or received by the agent.
- Read and save ACL messages from/to file.
- Sniff a particular agent (optional).
- Create ontologies graphically.

VMA also provides graphical interface for the administration of the Directory Facilitator and Agent Management System. Because VMA is an agent therefore it would communicate with AMS agent and DF agent through passing ACL messages. For the creation of ACL messages VMA package will use ACL package and will compose ACL message. After that the ACL message will be send to the Message Transport Service that will forward that message to the respective agent.

Chapter 4

GRID COMPUTING

This chapter gives an overall technological overview of Grid technologies along with the Semantic Grid. It also gives an account of the work already being done in this field.

The worldwide web has provided common grounds to share the information all around the globe. The Grid technologies build on this by allowing facilitating the global sharing of not just information, but also of physical resources (that are in terms of computational and data storage resources) to be used at a distance. E mail and WWW provide the vital means that allow communities that covers different states, countries and continents to work together in collaboration. Visualize a scenario if everyone could link their data, computers and other resources into a single virtual office .Grid technology basically enables this to make it possible by providing the protocols, services and software development kits needed to enable flexible, controlled resource sharing on a large scale.

At the core of Grid is the concept of virtual organization. It is a dynamic collection of individuals, institutions and resources bundled together in order to share resources as they tackle common goals. This resource sharing is not primarily file exchange, but rather direct, controlled (i.e. within the authorization, security, copyright, etc. restrictions) access to computers, software, data and other resources, as is required by a range of collaborative problem solving and resource brokering strategies emerging in industry , science and engineering.

Grid computing is an innovative approach that leverages existing IT infrastructure to optimize compute resources and manage data and computing workloads. According to Gartner, "a grid is a collection of resources owned by multiple organizations that is coordinated to allow them to solve a common problem." Gartner further defines three commonly recognized forms of grid:

- **Computing grid** - multiple computers to solve one application problem
- **Data grid** - multiple storage systems to host one very large data set
- **Collaboration grid** - multiple collaboration systems for collaborating on a common issue

4.1 Grid Services

Grid middleware should facilitate novel means to be build dynamically and transparently from distributed services. In order to engineer new Grid applications it is desirable to be able to reuse existing components and information resources and to assemble and co-ordinate these components in a flexible manner. Partly for this reason the Grid is moving away from a collection of protocols to a service-oriented approach: the Open Grid services Architecture (OGSA)[13][14]. This unites Web services with Grid requirements and techniques.

The Grid's requirements mean that Grid services extend Web services considerably. Grid

service configurations are:

• dynamic and volatile A consortium of services (databases, sensors, compute servers) participating in a complex analysis may be switched in and out as they become available or cease to be available;

• ad-hoc. Service consortia have no central location, no central control

• large. Hundreds of services could be orchestrated at any time;

• long-lived. A simulation could take weeks. .

These requirements make strenuous demands on fault tolerance, reliability, performance and security. Whereas Web services are presumed to be available and stateless, Grid services are presumed to be transient and stateful.

4.2 Semantic Grid

Until very recently the Grid and the Semantic Web communities were separate, despite the convergence of their respective visions. Both have a need for computationally accessible and sharable metadata to support automated information discovery, integration

and aggregation. Both operate in a global, distributed and changeable environment. The Semantic Web base services can be Grid Base Services. The Semantic Web fabric is the means by which the Grid could represent metadata: both for Grid infrastructure, driving the machinery of the Grid fabric, and its base and high level services, and for Grid applications, representing the knowledge and operational know-how of the application domain.

Semantic Grid is an initiative to develop effective methods for enabling such complex resource sharing. The key to this is an infrastructure where all resources, including services, are adequately described in a form that is machine-processable, i.e. knowledge is explicit - in other words, the goal is to provide semantic interoperability, based on the technologies of Semantic Web.

4.3 Semantics for Grid Computing

4.3.1 Semantic Grid services

The description of a service is essential for automated discovery and search, selection, matching, composition and interoperation, invocation and execution monitoring. This choice depends on service metadata. Classification of services based on the functionality they provide has been widely adopted by diverse communities as an efficient way of finding suitable services, e.g. UDDI. Reasoning over service descriptions has a role to play when classifying and matching services. In Condor a matching mechanism is used to choose computational resources. In an architecture where the services are highly volatile and configurations of services are constantly being disbanded and re-organized, knowing

if one service is safely substitutable by another is essential.

At the time of writing, the current state of describing Grid services through semantics is

by using the names assigned the portType and serviceType elements of a WSDL document, linked to a specification document. Bringing together the Semantic Web and

Web services have already attracted attention. DAML+OIL has been explored in myGrid.The myGrid service ontology extends the DAML-S ontologies. Service classifications are more expressive than UDDI's simple hierarchies and services are queried and matched by subsumption reasoning over the service descriptions. However, Grid services dynamically create and destroy service instances, have soft state registration and form long-lived service configurations. How this affects the way Semantic Web technologies can describe and discover Grid services is a challenge yet to be adequately addressed.

4.3.2 Integration of Information

Compound queries posed by scientists require the fusion of evidence from different, independently developed and heterogeneous resources. In biology, for example, the hundreds of data repositories in active service have different formats, interfaces, structures, coverage. The Web and the Data Grid guarantee a certain level of interoperability in retrieving and accessing data. The next level of interoperability is not just making data available, but understanding what the data means so that it can be linked in appropriate and insightful ways, and providing automated support for this integration process.

Information mediation is not restricted to traditional scientific databases. Computational resources are discovered, allocated and disbanded dynamically and transparently to the user. Semantic Web and Database technologies offer great possibilities. A common data model for aggregating results drawn from different resources or instruments could use RDF. Domain ontologies for the semantic mediation between database schema, an application's inputs and outputs, and workflow work items could use DAML+OIL/RDF(S). Domain ontologies and rules can be used for constraining the parameters of machines or algorithms, and inferring allowed configurations. Execution plans, workflows and other combinations of services benefit from reasoning to ensure the semantic validity of the composition. So we can use Semantic enriched services for:

- The classification of computational and data resources, performance metrics, job control; schema integration, workflow descriptions;

- Typing data and service inputs and outputs;
- Problem solving selection and intelligent portals;
- Infrastructure for authentication, accounting and access management.

Turning this around, we can envisage that the Base and Application services of the Semantic Web are implemented as Grid services.

Chapter 5

COMPARATIVE ANALYSIS OF FIPA-SL AND OWL

In this chapter my primary focus is to carry the in depth analysis of ontologies of OWL and FIPA SL .

OWL is a standard by W3C, which describes the ontologies for explicitly specifying the semantics and the distributed services. Its goal is for developing applications that utilize the semantically rich data processing and invocation of services that are described in OWL. As the OWL is based on description so it is the most suitable language for defining concept hierarchies and their relations, and is built on the top of RDF, which provides additional vocabulary and formal semantics. In the OWL ontologies we can specify the classes, subclasses there attributes, object properties etc.

FIPA SL on the other hand provides human readable string encodings (i.e. the contents of the FIPA SL are treated as a string) and is probably the most diffused content language in scientific community dealing with the autonomous software agents. FIPA SL provides certain subsets for imposing profiles that is, statements for imposing statements of restriction over the full expressive power of FIPA SL.FIPA SL is capable of representing propositions, actions and representing objects including identifying expressions to describe the objects.

The use of our proposed mappings/translations between the two languages will make it possible to describe the logical nature and context of the information being exchanged, while allowing for maximum independence between communicating parties and can be used to flexibly map specific data representations to these languages for eliminating the need to explicitly convert applications to adopt a certain data standard. Different agent based application will be able to use these mappings/translations to transform data from ontology to the other or to perform federated queries from a single query statement.

OWL Ontology	**FIPA Ontology**
SuperClass	Concept
SubClass(rdfs:subClassOf)	SubConcept
Object property	Slot
Restriction(has some values)	Predicate
Individual	Individual
disjointWith	NOT-SAME-VALUES
unionOf	Binary LogicalOp = "and"
intersectionOf	Binary LogicalOp = "or"
cardinality	CARDINALITY
minCardinality	MINIMUM-CARDINALITY
maxCardinality	MAXIMUM-CARDINALITY
someValuesFrom	SUBSET-OF-VALUES
InverseOf	SLOT-INVERSE

Table 1: Comparison of OWL and FIPA Ontologies

We compare the two languages in terms of expressive power and interoperability. We discuss the pros and cons when FIPA-SL or OWL is used as content language in ACL message structure and choose one language which will be better in terms of expressiveness in our the design and implementation for agent communication. Table 1 shows the detailed mappings of OWL and FIPA ontology.

We our focusing to carry out in depth analysis and view the tradeoffs between FIPA-SL and OWL when they are used as content language in ACL message structure. Logic, syntax and semantic of the both the languages are totally different from each other thus difficult to implement.

- OWL can be used as the content language witin an ACL message as to carry the contents of the message. Which has its other benefits and drawback to be used as content language in ACL message structure
- The FIPA-SL family of languages, which give a human-readable text representation for, messages which resembles LISP or scheme.

5.1 Expressiveness of OWL and FIPA SL

The FIPA-SL language is a human-readable string encoded (i.e. a content expression in FIPA-SL is a string) content language and is probably the mostly diffused content language in the scientific community dealing with intelligent agents. FIPA-SL is particularly indicated in open applications where agents from different developers and running in different environments have to communicate.

FIPA Content Languages should be expressive enough to satisfy the following requirements:

- They are capable of representing objects, including identifying expressions to describe objects.
- Proposition should be completely represented.
- They are capable of representing actions

According to FIPA SL an action expression expresses an action that can be performed. An action may be a single action or a composite action built using the sequencing and alternative operators. An action is used, as a content expression when the act is request and other communicative acts derived from it.

The propositions may not be always atomic, that is, they may involve explicit or implicit quantification, logical connectives like NOT, AND, OR and modal operators like BELIEF, DESISRE and INTENTION.

OWL as content language guarantees that the semantics of the messages exchanged is explicit and unambiguous between the sender and receiver. Owl has all the features needed to describe rich knowledge structures by agreeing on how meaning is conveyed it is simpler for applications to share meaningful content. The benefit of adopting a stronger semantically rich content language like OWL is that it facilitates a higher-level of interoperability between agents. But as OWL comes in three constructs (OWL-Lite, OWL-DL and OWL-Full) the layering structure of these three subsets causes the interoperability issues between them.

OWL expressive power as a knowledge representation language seems to be adequate for most needs of current agent based systems. OWL is also able to represent proposition, actions and objects. For examples this is how we can represent objects and actions in OWL

1. Maruf bike is excellent

   <Bike rdf:ID="Maruf_bike">

   <owner>Maruf</owner>

   <status> excellent </status>

   </Bike>

2. Maruf is eating Food

   <owl:Action rdf:ID="eating_Action">

   <owl:act>eating-food </fipaowl:act>

   <owl:actor>Maruf</fipaowl:actor>

   <owl:Action>

Fisrt statement is identifying an object and second is showing an action of eating in OWL. OWL has rich class, property, and axiom to model the world it can also express the logical connective like AND, OR, NOT and IntersectionOf, UnionOf and ComplementOf OWL constructors.

5.2. Representing Agents Attitude

The mental state of an agent is an intentional description making reference to beliefs, desires, intentions and other modalities that agents may have. In the agent world, to describe the mental state of an agent before sending (i.e., pre-conditions) a message and after having received it (i.e., post-conditions) is a useful way to ascribe meaning to communication primitives. The main challenge in this context is not only to provide a clear formalism for expressing the semantics in an unambiguous way for human developers but also to make sure that the agents' process of interpreting the meaning of the message is compliant to the given semantics. Pre-conditions and post-conditions are usually expressed in modal logic. The semantics of a communicative act (CA) is specified as a set of SL's formulae.

SL a quantified, multi-modal logic, with modal operators represented as follows:

- **Belief: (B <agent> <expression>)**

  Agent believes that expression is true

- **Uncertainty: (U <agent> <expression>)**

  Agent believes that expression is more likely true than false

- **Intention: (I <agent> <expression>)**

  Agent intends that expression will becomes true and will plan to bring it about

- **Persistent Goal: (PG <agent> <expression>)**

  Agents' holds a persistent goal that expression becomes true, but will not necessarily plan to it about

But the problem is that we cannot express these modalities in OWL as OWL is not based on Modal Logic so we cannot express the modalities like Beliefs, Uncertainty, Desires and Intensions etc. The agent mental attitudes about the state of the world cannot be expressed when OWL is used as the content language. Only SL is able to express these modalities. This is where SL distinguishes its expressive power from OWL

5.3. Comparison based on Logic

Logic is a system or model of reasoning. The simplest kind of logic called Propositional Logic deals with simple expressions that can be assigned a value of true or false. First-Order Predicate Logic extends Propositional Logic by allowing separate symbols for predicates, subjects and quantifiers. SL is based on First Order predicate logic and Modal Logic so the expressive power of SL includes rich set or operators and constructs but as the expressive power increases the complexity of reasoning decreases.

A description Logic is appropriate to use as a common core. Description Logics forms a family knowledge representation (KR) that represents the knowledge of an application domain. Description logics represent a subset of first-order predicate logic aimed at being tractable while maintaining a richness of semantic expressiveness. It has declarative formalism for the representation and expression of knowledge and sound, tractable reasoning methods founded on a firm theoretical (logical) basis. Owl is based on Description logic and is distinguished by its decidable characteristic and a formally defined semantics, which enables the subsumption relationship to be computed by suitable algorithms. Compared with many of its predecessors, Description Logics provides a formal, logic-based semantics to make it a useful knowledge representation framework for different application domains. The expressive power of OWL is not as much as SL as it is based on Description logic, which is a subset of first order predicate logic, but it is decidable and has reasoning capability.

5.4 Issues between interoperability of OWL and SL

OWL is widely used within the FIPA community, though not yet formally adopted as a FIPA compliant content language. OWL is a W3C recommendation so while using it in the ACL message structure the incompatibility lays between two major governing bodies FIPA and W3C. Also in the recommended agent platforms java is widely used in agent development and agent communication [7] therefore when ACL is programmed using java and content language is programmed using OWL the overhead of using two different languages for agent communication also increases. We can use OWL as a content language in ACL message structure but the operating cost increases and efficiency decreases. Unfortunately the different languages have slightly different semantics and are generally not interchangeable in the same agent (without modification anyway). The SL-encoding is the richest in terms of built-in operations and other rich constructs. FIPA-SL is the preferred language for the coursework as it is human-readable and has efficient encoding and parsing techniques.

After a comparative study of SL and OWL we found that SL had the following advantages:

- SL is well known to the project partners because it is widely used within the FIPA specifications both for the specification of the semantics of the FIPA Agent Communication Language, and for most examples in the whole set of specifications
- SL is relatively easy to learn and understand

- SL is expressive enough for the purposes of the agent communication and semantics

Chapter 6

PROPOSED ARCHITECTURE

This chapter introduces the overall architecture of the system along with the detailed communication architecture.

Evolution of World Wide Web has made it a focal point of research to apply structured and well-defined meanings to achieve Semantic Web and ultimately the Semantic Grid. The realization of this goal gives rise to the integration of technologies that can negotiate and cooperate in which agents have proven to be effective. OWL (Web Ontology Language) is currently the W3C standard for providing the explicit semantics to the web services whereas the FIPA Semantic Language is the core of agent platforms due to its high expressive power. The key objective of this paper is the development of semantic translations in such a way that the agents can communicate with the web services in an efficient manner. The challenging domain where our work can be applied is, the messages that can be exchanged between FIPA compliant Multi Agent systems and the web services, thus enabling the agents based applications to interact with the web services. Our goal for these mappings/translations is to show that how the existing standards of FIPA Semantic Language could be mapped into representations, which could be readily used in a Web Services, based environment.

Our devised model provides the semantic interoperability in distributed environments where technologies like agent applications and grid systems are combined and reused to achieve the autonomous semantic grid and thus providing a service oriented framework. The Semantic Web provides a conceptualization by which the distributed knowledge and be represented and viewed, in terms of formation, utilization, propagation and management. The concepts can be defined within extensible, open ontologies, which are published using the standard protocols with occurrence and properties being asserted at arbitrary locations across the web.

As we are specifying that the OWL will be the W3C standard for specifying the services on the web and will act as a content language. The communication infrastructure specified by FIPA permit the agents to communicate using any mutually comprehensible content language as long as it fulfills a few minimal criteria as a FIPA compliant content language [FIPA, 2003].

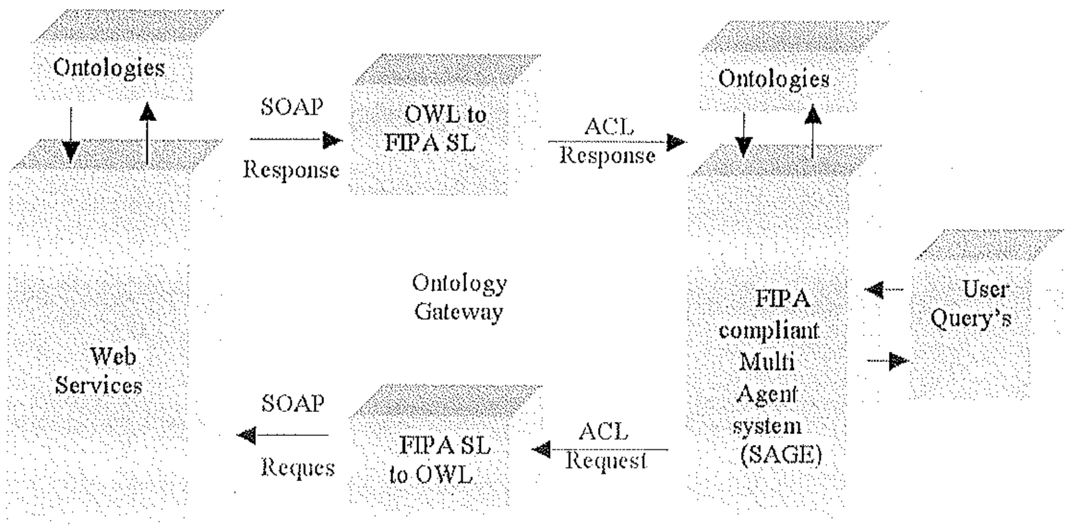

Fig. 6. Proposed System Architecture

In Figure 2 an abstract architecture of the proposed system is shown. The key idea is to show that how the software agents can communicate with the OWL based web services in bringing the semantic operability, negotiation etc. Another important aspect of the proposed system is that it is not reshaping the existing standards of the FIPA Semantic Language and Web Ontology Language. Our focus is the interoperability of the two languages so that the autonomous software agents can communicate with the web services or they can be used to build their knowledgebase. The middleware act as a converter, which by taking input data (from the ACL message) converts it into mappings/translations described in OWL and vice versa.

Major issues in realizing the proposed system were:

- FIPA SL and OWL have different underlying support for the terms, syntax, semantics and implementation constraints.
- The two languages are based on different standards one by FIPA and the other by W3C.

Devising mappings/translations for such a system is quite demanding and challenging, as both languages vary tremendously. Comparison to both the languages is provided in terms of representations and possible mappings in Section 3, which shows that the difference between the two languages made these mappings/translations very challenging.

This section describes the detailed design of proposed system in which the most important technical challenges are solved, i.e. without changing the specifications and standards of W3C OWL and FIPA SL, enabling the two-way communication and the content transformation entities including Grid services and Grid clients.

6.1 Software Agents communication with the OWL based Web Services

In this section, detailed design as shown in fig 3 that enables FIPA compliant Software Agents interact with W3C compliant OWL based Web Services.

A middleware is designed that makes services visible to Software Agents (see fig

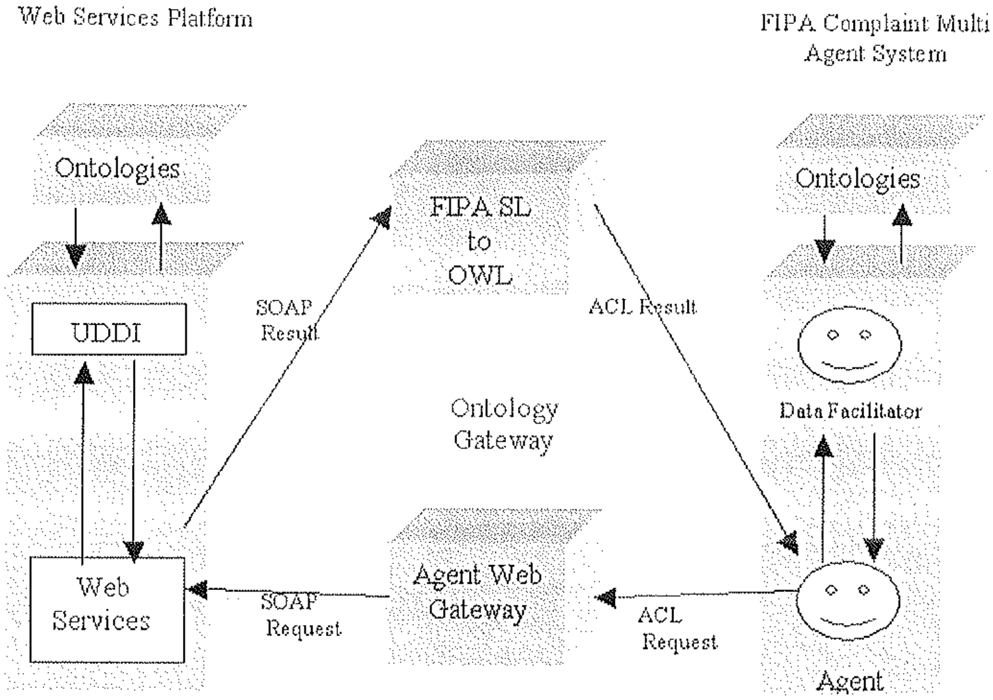

Fig. 7. Software Agents communicating with the OWL based Services

7).

Whenever an Agent needs to search for a service on the GRID environment the transformation of the ACL based search query into SOAP based UDDI search query and forwards to UDDI where the required service is expected is done using the Agent Web Gateway's component named ACL2SOAP converter [12].Now the Software Agent has come to know about the existence and the address of the required service. The address for the Web Service file is now retrieved and the Software Agent extracts the required Web Service. Software Agent is needed to know about the Ontology, AgentAction Schema, Predicate Schema and Concept Schema etc. so the Web Service is passed to Ontology Gateway that translates the Web service published in OWL into an equivalent FIPA compliant message.

6.2 Grid entities communication with the Software Agents

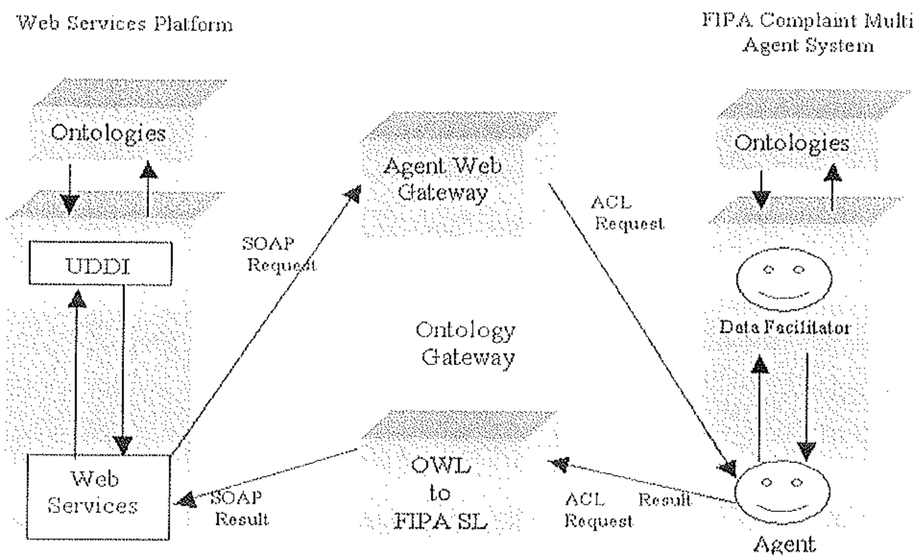

Fig. 8. Grid entities communication with Software Agents

In this section, the shown detailed design enables W3C compliant SOAP based Grid computing environment entities including Grid services and Grid clients interact with FIPA compliant Software Agents by performing Service Discovery in DF, understanding services provided by a Software Agent and consuming services from Software Agents.

A middleware is designed that makes services visible to Grid Entities (see fig 8). Whenever a Grid Entity needs to search for a service on the GRID environment the SOAP based UDDI query is passed to a protocol converter of Agent Web Gateway named SOAP2ACL. The protocol converter extracts out the service name sends a valid ACL based DF search request message to the Agent Platform. Directory Facilitator performs a search. If the required service is not found it is forwarded to a remote platform where the service is expected.

Directory Facilitator of the remote Agent Platform performs a search. If required service is found, the service is returned by the DF of that remote Agent Platform to the agent at our middleware, which converts the FIPA Ontology into the OWL Ontology. And the service retrieved is embedded into the original SOAP message, which is forwarded to the Web Service Client that requested for the search. In this way, the Ontology Gateway helps the Web Service client to search for the services at Agent Platform.

Chapter 7

CONCLUSIONS AND FUTURE DIRECTIONS

This chapter introduces conclusive remarks and future work that needs to be done in order to improve the efficiency of communication between agents and web services.

7.1 CONCLUSION

In this thesis we have devised translation/mappings for the interoperability of FIPA Semantic Language with the Web Ontology Language in the context of achieving a semantic web and ultimately the semantic grid to achieve autonomous coordination in the messages that are exchanged between the FIPA compliant Multi agent systems and OWL based web services. Our proposed system provides a robust infrastructure for semantic interoperability of messages between the two languages. We have done a detailed analysis of FIPA Semantic Language and the W3C OWL. The detailed design then explains the role of middleware that facilitates the two-way communication among the FIPA compliant software and W3C specified Web services without violating the existing standards. There are many applications that benefit from having a dedicated agent that manages and controls access to a set of explicit ontologies.

In information retrieval applications, the size of some linguistic ontologies may prevent an agent from storing the ontology in its address space, so that agents need to remotely access and refer to ontologies for disambiguation of user queries, for using information about taxonomies of terms or thesauri to enhance the quality of retrieved results, etc. The definition of a standard interface to access and query an ontology service can increase and simplify the interoperability between different systems.

Semantic integration of heterogeneous information sources in an open and dynamic environment, such as the Internet or a digital library, may also benefit from an ontology service. There are already implementations that use one domain ontology to integrate several information sources, managed by a dedicated agent, whilst still allowing each source to use its private ontology. Every user can also have their own ontology depending on their preference, their role in the domain or simply their known language. Every used ontology is a subset of the domain ontology or there exists a map between it and the domain ontology; the knowledge about these relationships (subset and mapping) is usually maintained by some ontology dedicated agents.

7.2. FUTURE DIRECTIONS

Rich semantic and pragmatics are the foundation for the high level negotiations among Multi-Agent systems and they can provide a better framework for negotiation among agents. Highly expressive content language and better encoding techniques attempts to facilitate the use to capture a wide variety of negotiation mechanisms in a principled and well-structured way. The translations can be further enhanced for OWL-S and similarly new operators can be introduced as to efficiently communicate between two languages.

RESULTS AND DISSUSIONS

8.1. OWL to SL Ontology Conversion

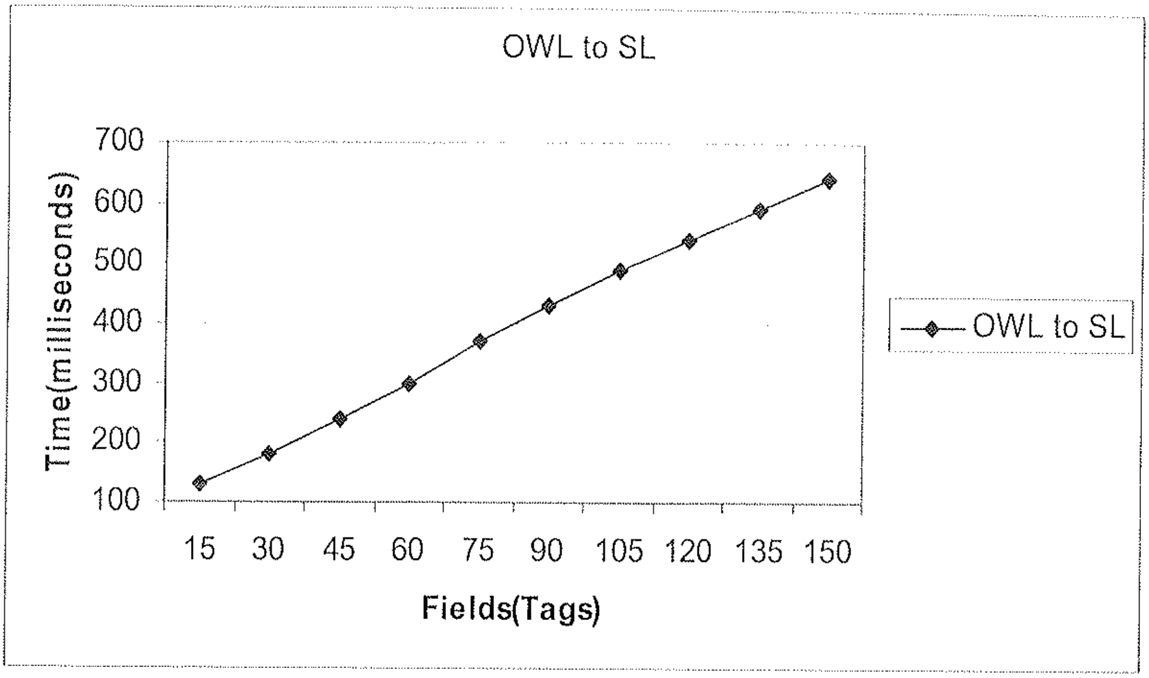

Fig. 9. OWL to SL Ontology Conversion

8.2. SL to OWL Ontology Conversion

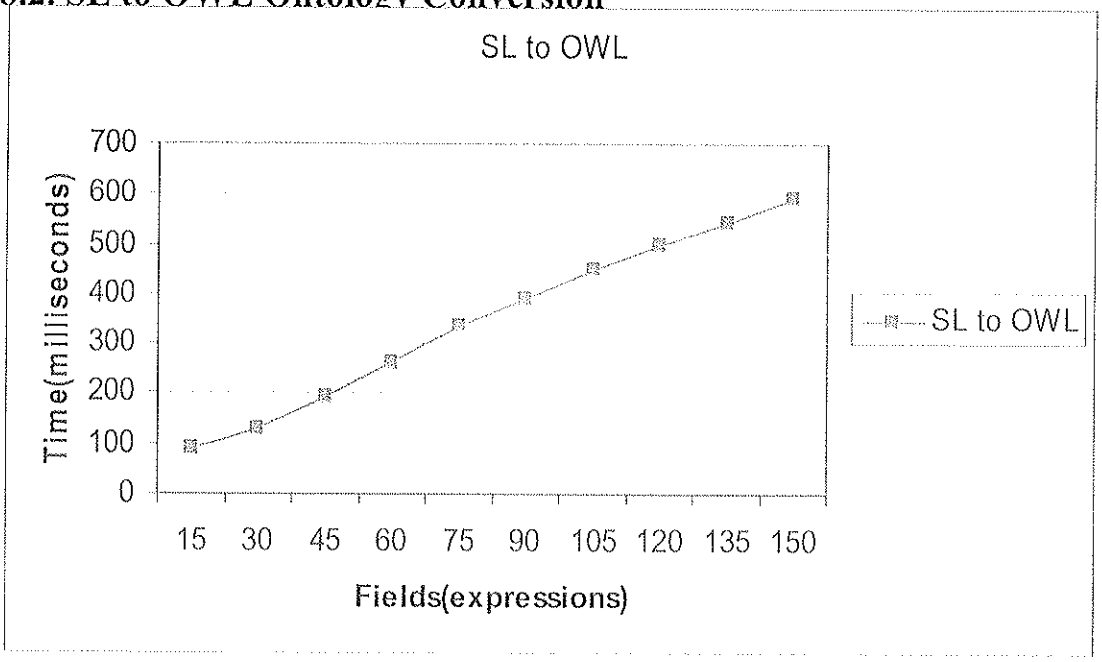

Fig. 10. SL to OWL Ontology Conversion

8.3. Ontology Conversion Comparison

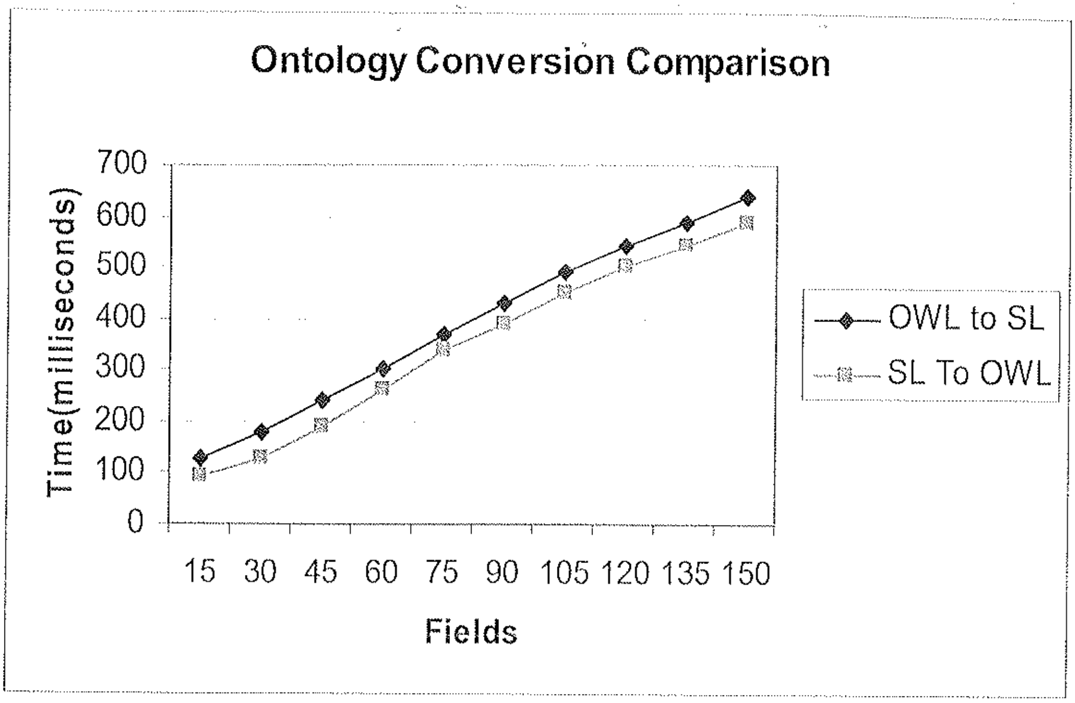

Fig. 11. Ontology Conversion Comparison

REFERENCES

1. T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. Scientific American, 284(5): 34-43, May 2001.
2. Horrocks I. Reasoning with Expressive Description Logics:Theory and Prac tice. In:Andrei Voronkov, (ed) Proc. 18th Int. Conf. on Automated Deduc tion(CADE-18).Springer Verlag,pp.1-15,July 2002.
3. J. Ferber. Multi Agent Systems, Addeson Wesle, ReadingMA, 1999.
4. D. Fensel. Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce, 2nd edition. Springer-Verlag, Berlin, May 2003.
5. F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, eds. The Description LogicHandbook. Cambridge University Press, 2003.
6. D. Brickley and R. V. Guha. RDF vocabulary description language 1.0: RDF schema. Recommendation 10 February 2004, W3C, 2004.
7. M. Dean and G. Schreiber, eds. OWL Web Ontology Language Reference. 2004. W3C Recommendation 10 February 2004.
8. M. Wooldrige, "Agent based Software Engineering IEEE Proc. Software Engineering, vol. 144, no .1, pp.26-37, March 1997".
9. H. Chen, Y. Chung, M. Ramsey, C.C. Yang, An intelligent personal spider (agent) for dynamic internet/intranet searching, Decision Support Systems 23 (1 41– 58)June (1998).
10. Zaheer Abbas Khan, H. Farooq Ahmad, Arshad Ali, Hiroki Suguri, China, April "Decentralized Architecture for Fault Tolerant Multi Agent System", ISADS, 04, 5-7, pp 167-174 April 2004.
11. Abdul Ghafoor, Mujahid ur Rehman, Zaheer Abbas Khan, H. Farooq Ahmad, Arshad Ali, "SAGE: Next Generation Multi-Agent System", PDPTA'04, pp.139- 145, Vol. 1,March (2004).
12. Muhammad Omair Shafiq, Hafiz Farooq Ahmad, Hiroki Suguri, Arshad Ali,"Autonomous Semantic Grid: Principles of Autonomous Decentralized Systems for Grid Computing" IEICE/IEEE TRANS., VOL. E85-A/B/C/D, No. 1(2005).
13. S. Tuecke, ANL; K. Czajkowski, USC/ISI; I. Foster, ANL; J. Frey, IBM; S.Graham, IBM; C. Kesselman, USC/ISI; T. Maquire, IBM; T. Sandholm, D.Snelling, Fujitsu Labs; P. Vanderbilt, NASA, GWD-R (draft-ggf-ogsi-gridservice-33)"Open Grid Services Infrastructure (OGSI) Version 1.0", June 27, 2003.
14. Ian Foster, David Snelling, "Web Service Resource Framework - WSRF", http://www.globus.org/wsrf/faq.asp, March 2004
15. The Web Services Agent Integration Project AgentCities.NET
16. C. Walton and A. Barker, "An Agent-based e-Science Experiment Builder", 1st International Workshop on Semantic Intelligent Middleware for the Web and the Grid, Valencia, Spain, August 2004.

APPENDIX

Test Bed Specifications

Machine Specifications

- Intel Pentium 3 (500 MHz)
- 256 MB RAM, 80 GB HDD
- Single PC
- Microsoft windows XP2004 Build 2600 Service Pack 1