Semantic Web Services in Action - Enterprise Information Integration

Parachuri Deepti and Bijoy Majumdar

Setlabs, Infosys Technologies Ltd., Bangalore
{Deepti_parachuri and Bijoy_majumdar}@infosys.com

Abstract. With the development and maturity of Service Oriented Architectures (SOA) to support business-to-business transactions, enterprises are using Web services to expose the public functionalities associated with internal systems and business processes. Semantic Web service infrastructure achieves automatic data integration to enable enterprises to collaborate and compete effectively in a dynamic global environment. In this paper, we deal with two important aspects of enterprise information integration, namely process integration and data convergence. This paper talks about solution strategies for global enterprise system which provides unified information and agile solution with greater ease and simplicity. Today's Web data lacks machine understandable semantics making it impossible to achieve data integration with the Web service. Hence, the semantic Web services in action to overcome the limitations of information finding, information extracting, information representing, information interpreting and information maintaining. This paper takes you through a case study simulating semantic Web paradigm (and semantic Web services) over a leasing business system. It also portrays the various advantages and explains the hurdles in accepting the semantic Web technology.

Keywords: Semantic Web, Web Services, RDF, OWL, OWL-S, Agent Technologies, SOA, Information Integration, Semantic Web Services.

1 Introduction

Machines cannot easily make sense of most of the information on the Web. Web data is chiefly designed for human consumption. Almost all metadata (e.g., HTML) describing Web documents is about where and how to present a piece of information. Many attempts have been made to automate and improve the gathering and use of information (by means of "spiders" and "wrappers") on the Web, but these technologies still only scratch the surface.

With the evolution of SOA and semantic Web services, automated processing and integration of data and application became easier. Externalization of atomic business capabilities is achieved through Web services by making the business interfaces transparent. Effective and automatic communication with in and between the organizations also raised the need for Web services. The mandate for the semantic data and Web services is the onset of distributed computing model SOA, to provide seamless integration not just for the services but also for the information sent across.

B. Krämer, K.-J. Lin, and P. Narasimhan (Eds.): ICSOC 2007, LNCS 4749, pp. 485–496, 2007.

Current business scenario needs a global enterprise system which provides unified and required information with a greater ease and simplicity. And also should be able to cater the requirements of a constantly changing environment (business environment changes, user requirement changes and technical environment changes etc.) which is the major drawback in traditional data integration systems [1, 2, 3] and data warehouses [4, 2]. Many organizations use Web services for managing distributed applications, such as health care, agricultural management system, insurance claim processing, etc.

Using software as a service helps in sharing of resources in the constantly sharing environment. Web Services (WS) evolved as a solution for publishing, discovering and invoking the software component as services. WS help in integrating interoperable distributed heterogeneous Web services. WS and Service-oriented architecture (SOA) are emerging distributed computing paradigms and are well suited for enterprise information integration. SOA is software architecture which provides interoperable integration of scattered services by using services as components. WS are based on standard internet protocol like XML for data representation, WSDL for binding and to define interface, UDDI for discovery and SOAP for message exchange and are accessible with the help of wide range of computing devices [5, 6]. Beside these advantages it also has drawbacks such as standards and specifications are syntax based, not matured enough, not machine process able, and not sufficient for certain kinds of applications where composition, security, state, transaction management and scalability are highly recommended [7,8].

Semantic [10] is a solution for finding meaningful information and integrating with related information. Semantic approach helps in searching, discovery, selection, composition and integration of WS and also in automation of invocation, composition and execution of services. Ontology [9] is the key technology behind Semantic Web for making information more meaningful, by adding more knowledge. Rules are the next development area in semantic Web to specify declarative knowledge, constraints and to enforce policy. Delivery of personalized context and location based information [11].

The major motivation behind this paper is to organize data of an enterprise in a well defined manner thereby enabling the machines to understand, interact and retrieve the content with a greater ease. In our work, we present a case study on *Leasing Business Enterprise* wherein we model the data using semantic technologies, RDF and OWL, at the *data level*. Next we model a *service level* using OWL-S, which can be processed by machine automatically. Interaction between services is provided using agent technologies.

2 Gaps Resulted by Distributed Computing

Many traditional solutions are available for information integration. One such solution is shown in Fig. 1. An enterprise consists of various processes and when the communication is between few processes then the existing framework catered the needs of the enterprise. But with the advent of business and technology, the need to interact with customers became important, existing solutions failed to handle the complexity in real time. Some of the shortcomings of the existing solutions are mentioned below.

- Integration is performed in the applications
  - Embedded, peer-to-peer integration
  - No reuse
- The integration process is heavy
  - Low reactivity to changes in requirements and processes
  - Low reactivity to changes in data sources
- Information is locked into proprietary formats
  - Difficult to integrate with external applications
- Integration is performed asynchronously
  - Out-of-date data
  - No access to operational data

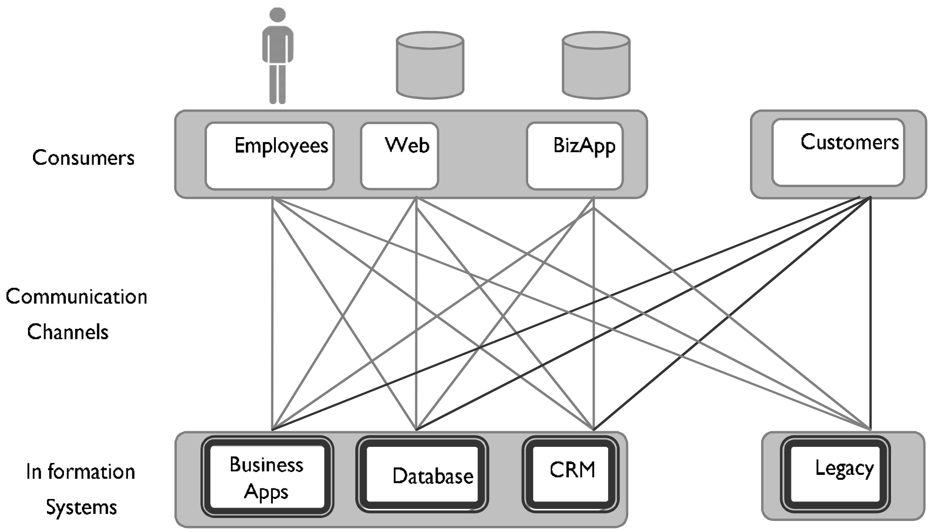

Fig. 1. Information Exchange across the Enterprise

About 80% of the Information Systems in Production Suffer from the following drawbacks

- Require IT assistance for end-user access
- Difficult for end users to identify relevant information
- Overload of information delivery
- Provide only partial answers to questions
- Often present out-of-date information
- Contain enormous amounts of redundant information
- Expensive to develop and maintain

We were delivering data but not information which can be consumed directly with out requirement of any processing.

The architectural framework of the proposed information integration as shown in Fig. 2 adopts a mediated ontology approach to data integration in which each data source is described by its own ontology and translations between different ontologies are by means of mediation.

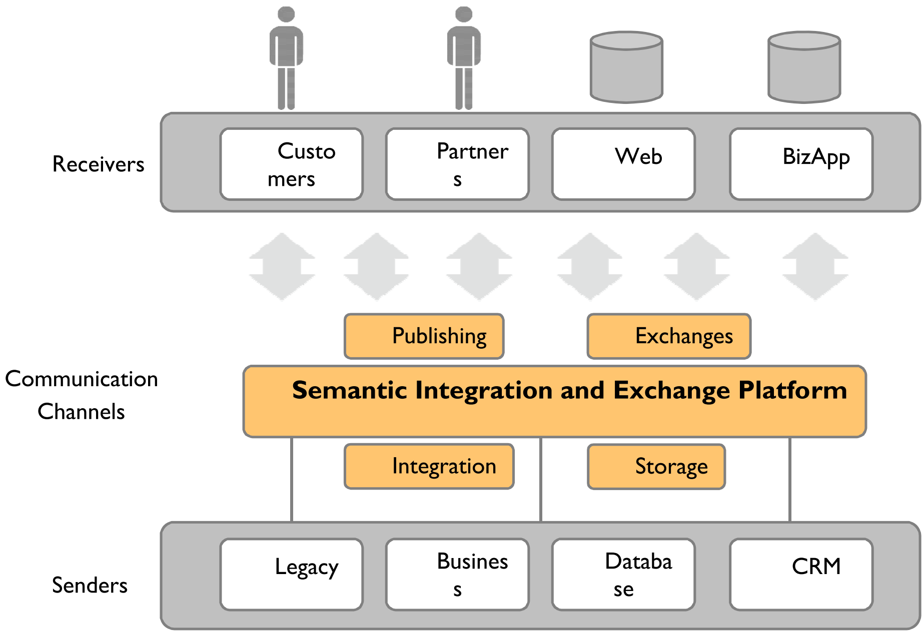

Fig. 2. Semantic based Enterprise system

3 Semantic at Data Level

3.1 Introduction

The semantic Web service has the potential of becoming the most powerful technology for information integration. It deals with two important and complementary aspects of information integration, namely data integration and service integration for effective discovery, automation, integration and reuse across various applications. Automatic data integration on a global scale is important for enterprises operating in a dynamic global market. The large number of data sources needed to access for an application and the changing business requirements make manual integration of data infeasible. With the use of Web services the data is made available for public access. But today's Web is human interpretable not machine understandable. Web data lacks machine understandable semantics making it impossible to achieve data integration with the Web service. Hence, the semantic Web services in action to overcome the limitations.

The idea of the Semantic Web is to refine the existing Web incrementally, inserting machine-readable "semantic" tags into Web documents or other data-streams. These tags are supposed to provide more information regarding the concepts within the data and their relationships to each other. The implications of such added semantic information could be far-reaching: Rather than being restricted to the Web, it would encompass virtually every aspect of life. “The two major business benefits are the promise for **tremendously improved search capabilities** and — in the long term — **improved systems interoperability**, potentially enabling machines to reach new levels of automation.” [Berners-Lee,2001] Such semantic tags will be increasingly used across many domains, but whether this will stretch across the whole Web in the near- to mid-term is still uncertain.

3.2 Advantages

The key standards for the development of semantic Web are RDF, RDF-S and OWL. Many resemblances have been noted between RDF and ER diagrams. When entities can be represented by URIs, RDF makes ideal candidate for storing ER diagram as machine readable text. RDF is actually more flexible than classical ER diagrams, because in RDF we can make one of the relationships that is one of the predicates as represented by its URI the subject or object of triples. The ability to treat predicates as first-class objects provides advantages. By using the W3C’s Web Ontology Language (OWL), equivalences between predicates makes it easier to combine databases without revising one database to have the same schema as the other. For example, if product_id and product_code are defined as equivalent, a search on products with product_id value of 101 will also get the details written under product_code value 101. This feature of RDF is an attractive approach to aggregating distributed data not controlled by a central authority. If we can define ontology to manage data then RDF triples are the best way to track entries into the ontology.

3.3 Challenges

Semantic technologies drive business value by providing superior capabilities (increased capacity to perform) in five critical areas [14]:

- **Development** — Semantic automation of the “business-need-to capability-to-simulate-to-test-to-deploy to-execute” development paradigm solves problems of complexity, labor-intensively, time-to-solution, cost, and development risk.
- **Infrastructure** — Semantic enablement and orchestration of core resources for transport, storage, and computing helps solve problems of infrastructure scale, complexity, and security.
- **Information** — Semantic interoperability of information and applications in context, powered by semantic models makes “killer apps” of semantic search, semantic collaboration, semantic portals and composite applications.
- **Knowledge** — Knowledge work automation and knowledge worker augmentation based on executable knowledge assets enable new concepts of operation, super-productive knowledge work, enterprise knowledge-superiority, and new forms of intellectual property.

- **Behavior** — Systems that learn and reason as humans do, using large knowledge bases, and reasoning with uncertainty and values as well as logic enable new categories of hi-value product, service, and process.

4 Semantic at Service Level

4.1 Introduction

The major motivation for using Web services is to reduce cost, effort and time in integrating enterprise applications but Web service usability and integration needs to be inspected manually. There is no semantically marked up content / services. Only syntactical descriptions are present. Hence requires people to locate services and create interfaces. Semantic Web Services emerged as integrated solution for realizing the vision of the next generation of the Web. Service ontologies provide a way to automatically integrate and manage the integration thereby reducing the total cost of integration.

DAML-S (OWL-S) is a DAML+OIL-based Web service ontology, which supplies Web service providers with a core set of markup language constructs for describing the properties and capabilities of their Web services in unambiguous, computer-interpretable form. DAML-S markup of Web services facilitates the automation of Web service tasks including automated Web service discovery, execution, composition and interoperation. In particular, it provides language primitives for technical, business-related and process-based facts about services. Thus, DAML-S can be regarded as a semantics-based substitution of the above-mentioned Web service languages for service description, service publication, and service flow.

4.2 Advantages

Semantic Web would provide greater access to not only content but also services on the Web. Users and software agents can discover, invoke, compose and monitor Web resources offering particular services and properties.

4.3 Challenges

A service ontology language should enable the following tasks

1. Automatic Web service discovery
2. Automatic Web service invocation
3. Automatic Web service composition and interoperation
4. Automatic Web service execution monitoring

5 Case Study

5.1 Overview

The case study used here, to depict the various challenges and semantic strategies, is the process driven in leasing business for a multi region and multi vendor system.

Fig 3 shows the process flow across various departments and systems that not only spans the organization but also other vendor entities. Applications that need to merge or synchronized with other exiting applications in different administrative domains require complying with the semantic platform to have a robust and agile business system. The lease process is defined in brief in the figure with the various business products and data being passed across systems.

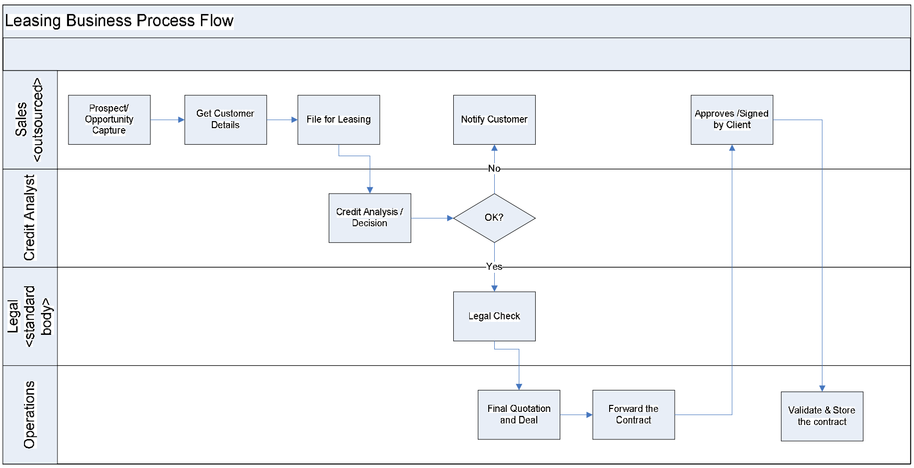

Fig. 3. Part of flow diagram of Leasing Business

5.2 AS-IS → TO-BE

The purpose of this exercise is to transform the manual operated leasing business with discrete applications carried out in discrete departments to automated business process with optimal manual intervention maintaining business agility. Up to date, million dollars are spent to develop enterprise architectures as a basis for IT modernization have largely resulted in manual compliance exercises, producing reference documentation, disconnected from operations and management systems, and delivering no capability to business users (which is the AS-IS scenario). The goal is to provide a semantic based integration platform which avoids manual indulgence to an extent and provides a flexible and agile business process (which is termed as the TO-BE scenario).

5.3 Information Perspective

In this section, we present a case study on leasing business to provide a deeper insight for designing of ontologies and conceptual modeling. It also aims at providing solution strategies for global enterprise system. Information integration provides benefits and challenges for different domain and application areas. In order to have synchronization in data representation in various vendor system or for any future acquired business, data convergence and knowledge management is a challenge that needs to be tackled. This will give way to many data format or data nomenclature differentiation that occur due to various administrative / ownership domains involved. Few key data convergence strategies and its significance are mentioned in the following sub sections. Each key area have enhanced the information definition and helped the leasing system.

Enterprise data integration: RDF

Enterprises comprises of many processes for e.g. Leasing business consists of many processes like sales, order processing, product catalog, credit analyst, Hr etc. Different departments need to share data, but the lack of an interoperable, integrated solution prevents this. Even if the companies want to cooperate with partner companies to exchange data across applications, the need for compliance to emerging standards and government regulations arises. Another scenario where the need for standard data format arises is when there is a merger or acquisition where the disparate software infrastructure and underlying content and functions of two companies need to be integrated. Fig. 4 shows a domain ontology stored in RDF format and provides mappings between various processes.

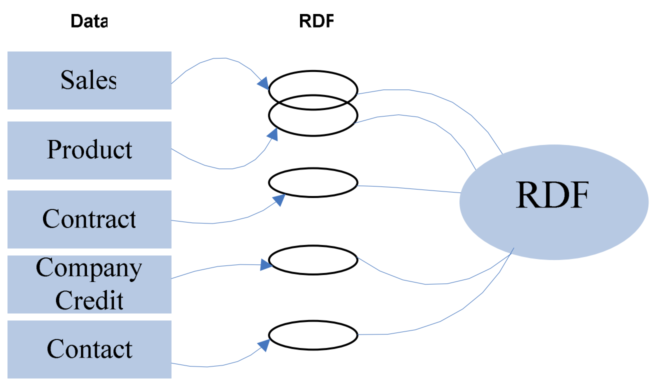

Fig. 4. Data integration within a Leasing Business enterprise application environment

Data Aggregation

An aggregator is an entity that collects and analyzes information from different data sources. Aggregation defines a new landscape in information retrieval for goods and services on the Internet. Aggregators provide access to comparisons of information and pricing that have not been possible in the past. In addition, after-aggregation information provides tremendous market intelligence whose value has yet to be realized.

Different data providers use different ways to structure their data, they use different identifiers to reference the same entities, there is acronym collision between the data sets. Even data is present in different formats namely file formats, XML schemas and relational models. RDF comes out as a better solution to overcome these problems. Fig 5 shows an example where prospect id and prospect code are alias names and refer to the same thing. RDF data format makes it easier to store the alias names and retrieves the data from both the processes if the query is “Get all details of prospect with prospect id: 10”.

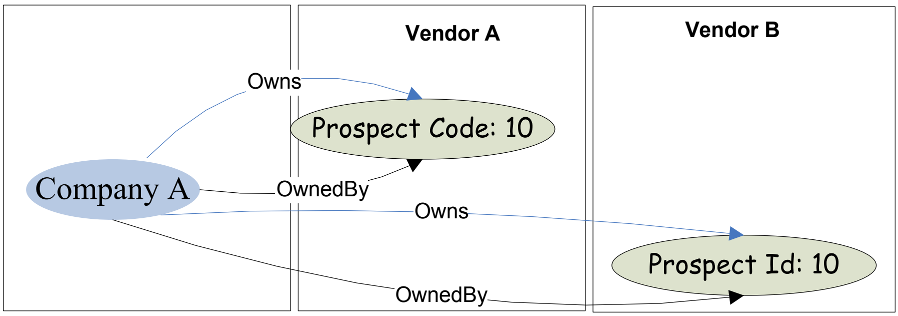

Fig. 5. Prospect Id alias Prospect Code

Content Aggregation

Content aggregation is done thorough Web 0ntology Language (OWL). Owl is for processing and interpreting the content on the Web.

Enterprise Search

The increase of both published and internal information presents a challenge in enterprises. Traditional search-based methods are unable to find relevant information in the required time scales.

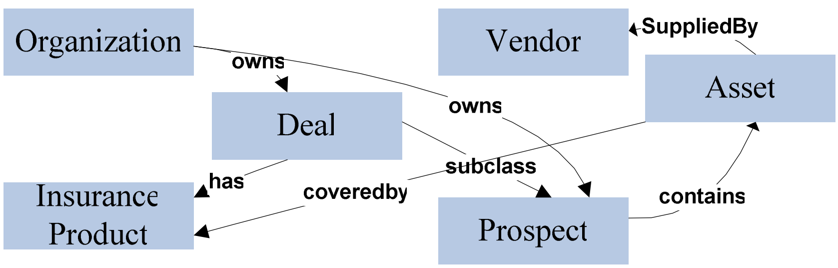

Fig. 6. Graph representation of RDF

As every relation in RDF is binary search becomes easier. Data is RDF is stored as directed graphs.

Managing Grid Resources

RDF refers to Resources, identified by URLs. This means that information about a single resource can come from many sources. Hence, having distributed data is easier in RDF format.

Mapping of ER diagram to RD

Most of the enterprises store data in relational model databases. It is not possible to model the data in RDF format from scratch, hence need for conversion of relational model databases into RDF formats arises. Fig. 9 shows one such conversion taking a part of ER diagram from leasing business using the method proposed in paper [13].

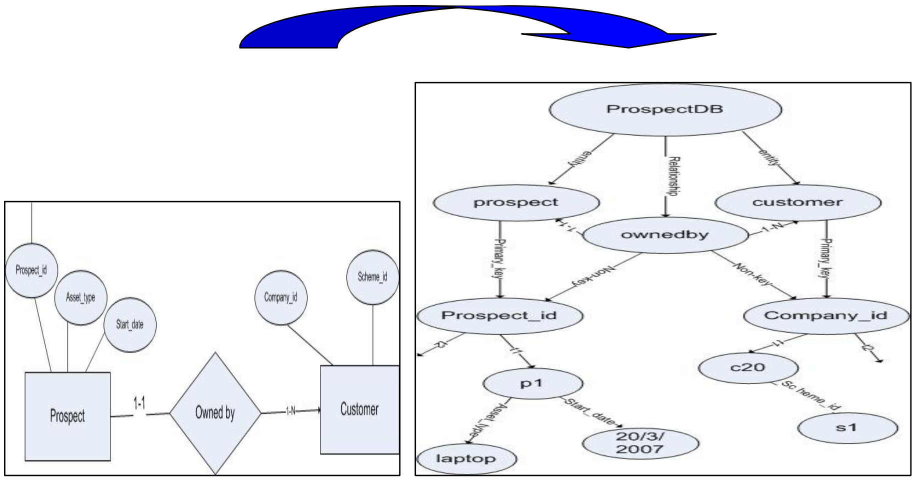

Fig. 9. An RDF equivalent of an ER diagram

5.4 Process Perspective

Once data or information has been streamlined to be understood by machine and across systems, now comes the part of automating the process to execute without any minimal manual intervention and smooth operation with available resources. As mentioned in earlier sections, OWL is a language that defines the best possible flow for a process enabled with knowledge and feedback system. But to productize such scenario in an optimal manner, software agents can come handy to choreograph the whole process.

Today the business market is buzzing with lots of BPEL standard based process engine. BPEL coupled with rules engine provides most automation with auto decision making and direction change in the flow. In addition to the system mentioned, embedding a grid engine is another option to pick the most optimal and best-of-breeds service.

Most of the other integration platform supports Service Component Architecture, which by and large, emulates the choreography strategy of process flow. This solution does not depend on one centralized component or a conductor to decide the route of a process but mediates through the services by intelligence and knowledge derived or calculated during the course of a process execution.

5.5 Positive Thought

The semantic EI approach includes semantic auto-discovery and mapping of legacy IT artifacts and documentation. This gives visibility and eliminates cost of as-is modeling, compliance auditing, and maintenance projects. Semantic discovery applied to IT artifacts gives the capability to scan source libraries, data schemas, and documentation, comments, etc. in order to identify unique artifacts, link and map dependencies, and do latent semantic indexing of the "as-is" world. The result is a repository of metadata (RDF/OWL), a very flat ontology that enables semantic (concept) search using business terms, without having to know the (often cryptic) as-built naming established by programmers. This could be thought of as a sort of "Google for IT" process that works bottom up, and also allows mapping linkages to enterprise architectures, or other governing models.

6 Assisting Technologies

6.1 Data Level

We have used RDF for data modeling and OWL for creation of ontologies. SPARQL is used as query language. Many publicly available editors are present for ontology creation and data modeling like

- **Protégé:** free open source editor and knowledge base framework.
- **Webnoto:** a tool providing Web-based visualization, browsing and editing support for developing and maintaining ontologies and knowledge models specified in OCML (an operational knowledge modeling language).
- **OilEd:** an ontology editor allowing the user to build ontologies using DAML+OIL. For further details and information about DAML+OIL.

We have used protégé to implement our case study.

6.2 Service Level Languages

We have used OWL-S to implement service level ontologies. We have used OWL-S plugin in protégé editor to model the ontologies. BPEL4WS has been used for service orchestration. Structurally, a BPEL4WS file describes a workflow by stating whom the participants are, what services they must implement in order to belong to the workflow, and what are the various orders in which the events must occur. The BPEL4WS process model is built on top of the WSDL 1.1 service model and assumes all primitive actions are described as WSDL port Types. That is, a BPEL4WS description describes the orchestration of a set of messages all of which are described by their WSDL definitions.

7 Future Work

We plan to use agent-based workflow management for service orchestration. The emergence of Web services and semantic Web facilitate the modeling of agent based system. In effect agent-based technologies provide the mechanism for components to seek work, enter into cooperative agreements and thus otherwise address the requirements of dynamic, heterogeneous environments.

8 Conclusions: Removal of Human Agents

The vision of Semantic Web has been to enable computer software to locate for us, relevant resources on the Web and also extract, integrate and index the information contained in the resources. Basically, make computers work on our behalf that is removing the human agents. In this paper, we have partially shown how information integration can be done in an enterprise using Semantic Web Services. We have used publicly available protégé tool to model the ontologies and store data in RDF Schema and OWL format. OWL-S is used for service aggregation and SPARQL is used for querying the data. We would like to extend our work by using agents for service aggregation and negotiation.

References

[1] Sheth, A., Larson, J.: Federated database systems for managing distributed, heterogeneous and autonomous databases. ACM Computing Surveys 22(3), 183–236 (1990)

[2] Jakobovits, R.: Integrating Autonomous Heterogeneous Information Sources (1997)

[3] Raman, V., Narang, I., Crone, C., Haas, L., Malaika, S., Mukai, T., Wolfson, D., Baru, C.: Data Access and Management Services on Grid (2002)

[4] Franconi, E.: Introduction to Data Warehousing

[5] Newcomer, E., Lomow, G.: Understanding SOA with Web Services, Addison Wesley, Reading (2004)

[6] Papazoglou, M.P.: Service-oriented computing: concepts, characteristics and directions. In: Proceedings of the Fourth International Conference on Web Information Systems Engineering, pp. 3–12 (December 2003)

[7] Birman, K.: Can Web Services Scale Up? IEEE computer (October 2005)

[8] Wang, H., Huang, J.Z., Qu, Y., Xie, J.: Web Services: problem and future directions. Journal of Web Semantics 1(3), 309–320 (2004)

[9] Uschold, M., Gruninger, M.: Ontologies: Principles, methods and applications. Knowledge Engineering Review 11(2), 93–115 (1996)

[10] Stuckenschmidt, H., van Harmelen, F.: Information Sharing on the Semantic Web. In: Advanced Information and Knowledge Processing, Springer, Heidelberg (2005)

[11] Laliwala, Z., Sorathia, V., Chaudhary, S.: Semantic and Rule Based Event-driven Services-Oriented Agricultural Recommendation System. In: ICDCSW'06. Proceedings of the 26th IEEE International Conference on Distributed Computing Systems Workshops, IEEE Computer Society Press, Los Alamitos (2006)

[12] Kabbaj, M.Y.: Strategy and Policy Prospects for Semantic Web Services Adoption in US online travel industry. MS thesis Submitted to Masters of Science in technology and policy at MIT

[13] Krishna, M.: Retaining Semantics in Relational Databases by Mapping them to RDF. In: IEEE/WIC/ACM (2006)

[14] Davis, M.: Semantic Wave: Executive guide to billion dollors