Intelligent Web Services System Based on Matchmaking Algorithm

Okkyung Choi[1], Seong Hwan Moon[1],
Department of Science & Technology Education for Life[1]
Seoul National University of Education[1]
Seocho-dong, Seocho-gu, Seoul, 137-742[1]
ok1124@freechal.com[1], hmoon@snue.ac.kr[1],
Korea

Sangyong Han[2], Ajith Abraham[2],
School of Computer Science& Engineering[2]
Chungang University[2]
Huksuk-dong, Dongjak-ku, Seoul, 156-756[2]
hansy@cau.ac.kr[2], ajith.abraham@ieee.org[2]
Korea

Abstract: - Web services, the key technology indispensable for e-business, is a comprehensible service providing the desired information or service in the same environment regardless of time and place by integrating current application systems within a single business or between multiple businesses with standardized technologies through the open network using the Internet. However the current web services search systems, based on the text oriented searching, are incapable of providing reliable search results by perceiving the similarity or interrelation between terms. Hence, currently no web services retrieval model containing such semantic web functions exists. This research is purported for solving such problems by designing and implementing an Intelligent Web Services System that is capable of searching for general web documents, UDDI and semantic web documents. The research proposes a Matchmaking Algorithm and verifies its efficiency and accuracy.

Key-Words: - Semantic Web Services, DAML-S, UDDI, RuleML, Matchmaking

1 Introduction

The most desired future World Wide Web would consist of Semantic Web providing more accurate and reliable information and Web Services that provide users with improved high-quality service by automated methods instead of only providing simple information. Such technologies pursue a conversion to decentralized services that support heterogeneous environments. Current Semantic Web Services model DAML-S also has some disadvantages in supporting automated web services. First, the model does not use the appropriate method for expressing the information, in short, the restriction conditions, and the user's requests are not applied sufficiently. Second, measurements of the service quality (fusibility, integration, performance, security and reliability) are inadequate[4].

As a solution to such problems the present research proposes the extended web services searching system. This system is differentiated from the existing Semantic Web Services searching system in the following aspects.

First, the meaning of the search keyword used within the domain is defined accurately by constructing an ontology server. Additionally, based on the constructed ontology, similar words are searched for and interpreted. Second, an extended UDDI search module is provided in order to solve problems with the existing UDDI. For this purpose, a matchmaking engine using the matchmaking algorithm was designed in this study. Third, rule-based searching is executed in order to reflect the users' preferences. Rule-based searching is a method for converting rules in the user profile registry and each returned web service result into Jess script through XSLT. Fourth, ranking is executed for more accurate information searches. The newly developed ranking algorithm was applied to rank search results and provide more accurate and reliable information.

The paper is organized in the following order. In Section 2 there will be an overview on the Semantic Web Services including DAML-S, Jena API and RDQL. In Section 3, we analyzed the problem of the current Semantic Web Services Model along with the solutions. In Section 4, the Matchmaking Algorithm is described and in Section 5, the architecture and implementation of improved Intelligent Web Services System is explained along with designing method, functions and characteristics of its modules. Conclusion will be suggested in the final section.

2 Literature Review

This Section presents an overview of the Semantic Web Services, Jena API and RDQL. The existing Semantic Web Services System is also compared and analyzed.

2.1 Semantic Web Services

Semantic Web Services which is the method purported for providing services for the users' convenience in the semantic web environment, is composed of SOAP, WSDL, UDDI and DAML-S. SOAP used for calling web services is a quantitative protocol for sending data that is structuralized and standardized using XML. However, it is also a type of message format protocols, which is highly extensive and can actually be applied in many other fields other than that of a quantitative protocol. WSDL for web services description contain information on the interface provided by the web services and the type of data that is to be used through the services. Despite many other related standards such as WSFL or RDF, presently WSDL plays the major role and is very highly extensive being capable of describing any type of sub-network or messaging protocol. UDDI is a directory service that can be used by an individual or company for registering and searching web services. In other words, UDDI allows one to register the service he/she is to provide on the web and the service user to find the service he/she wants from the registry.

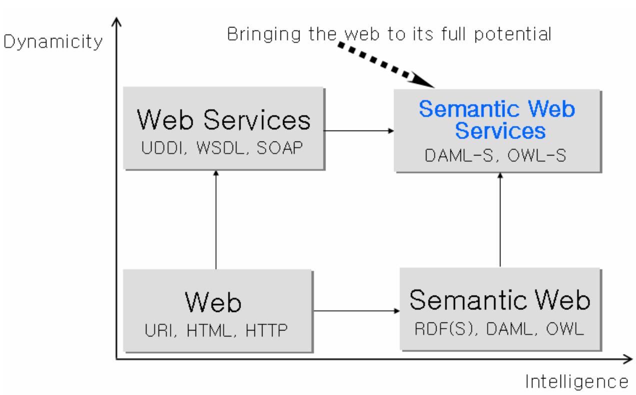

Fig. 1. Semantic Web Services

DAML-S(Semantic Markup for Web Service) is a key component for implementing Semantic Web Services, and at the same time, the ontology for web services. Service class, one of the elements composing DAML-S, is placed in the top class/layer and is provided by the Resource. Service is composed of Service Profile ("What does the service require of the user(s), or other agents, and provide for them?"), Service Model ("How does it work?) and Service Grounding ("How is it used?").

As so, within the Semantic Web Services, the DAML supporting agent reads the service descriptions in WSDL format, sends the information to DAML and then DAML connects to the appropriate ontology and the information is provided through the search engine. The advantage of this is that the service consumers can search for and link to providers of the desired web service through any search engine supporting such a framework and in addition personalization related information and negotiation abilities are allowable. In result, the agent is capable of referring to the consumer and provider information and perceiving the conditions in order to automatically settle contracts through negotiation.

2.2 Jena API and RDQL (A Query Language for RDF)

Jena is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS and OWL, including a rule-based inference engine. Jena is open source and grown out of work with the HP Labs Semantic Web Program. The Jena Framework includes RDF API, RDF, OWL API, RDQL(A Query Language for RDF)[3].

RDQL(A Query Language for RDF) has been implemented in a number of RDF systems for extracting information from RDF graphs. RDQL is an SQL-like syntax for this query model derived from SquishQL and rdfDB. Following lists are syntax and example of RDQL[3,10].

2.3 Analysis of the Current Research Works

InfoSleuth[1] is an agent based information search system, which uses the "broker agent" in order to execute syntactic and semantic matchmaking. The broker agent supports connections between service providing agents and holds recent information in the repository and the broker provides query agents, which locate all possible agents, in order to provide the appropriate service. The specific methods used here are syntactic brokering and semantic brokering. Meanwhile, InfoSleuth carries out services applying rules of LDL++ [13], logical deduction language. This language is not taken into account in this study since it is not a standardized service description language or a design method based on semantic web services.

In [5], a semantic web services system using OWL-S based brokers, uses agent based brokers for efficient web services execution and synchronization is presented. In this research, a new OWL-S is suggested in order to clarify the broker's functionality. The broker architecture shown in Figure 2 uses the Query processor and Discovery engine in order to provide service requesters with efficient and accurate service advertisement. However, no specific design plans are suggested nor any descriptions of matchmaking and brokering. Instead, descriptions mainly cover only the theoretical factors such as abstraction and pruning algorithms.

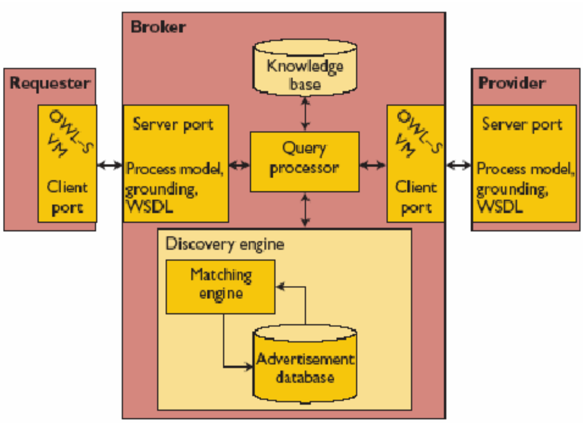

Fig. 2. The broker architecture

As described in [9], which is composed of DAML-S based on semantic web services description, the standard for matching web services search results is whether the service requester's request and the service provider's advertisement match or not. The matches are made by comparing each of the service inputs and outputs and classified into four matching levels: Exact, Plug-In, Subsume and Fail. This service grading method using the matching algorithm enables efficient web service searching but not capable of drawing accurate service search results because the matching results of each level are not ranked in detail.

3. Semantic Web Services Model

3.1 Problem Analysis

Based on web services related quality evaluation factors, this section will deal with the problems with web services in the semantic web environment. The problems can be described largely in four aspects as outlined below.

(1) User's Request Processing
The basic web services methods including UDDI are simple search methods such as syntax analysis that do not apply semantic factors. Furthermore, services, such as the artificial intelligent agent, which describes the information desired by the user and compare the search results are not provided. As a result, it is difficult to verify and confirm the user's requests.

(2) Composition of Services
For integration and composition between unit services, separate ontology information must first be integrated. However the current web services framework does not take care of this sufficiently. Web composition languages used at present include XLANG suggested by Microsoft, WSFL (Web Service Flow Language) by IBM and BPEL4WS (Business Process Execution Language for Web Services) suggested by Microsoft and IBM together. However, because these languages are all different standardization processes, they rather function as obstacles to environment integrating factors. Furthermore, there is a need for an automated web services composition technology the same as the agent technology.

(3) Service Quality Measurement
The most important factor of the semantic web service is how much the user can trust and rely on the service provided. However, other web services methods currently used do not process reliable messages and the waiting time, message processing and transferring time is too long resulting in poor performance.

(4) Satisfaction with Search Results
The current web services search methods do not apply priority ranking of search results or automated classification methods. A ranking method that considers users' requests and preferences is greatly required.

3.2 Solutions

(1) User's Request Processing
Problems with the simple searching method can be solved by the E-engine Ontology Server [2]. In order to apply semantic factors, search for groups of analogous terms is requested and the ontology system returns semantic information of the search results.[2] Additionally, the user's request can be verified and applied by using the Search Manager [2] suggested in former studies and executing repeated queries for

analogous terms, and the current UDDI's simple search method can be improved to enable efficient matching between service requests and advertisements by using the matchmaking engine.

(2) Composition of Services
For interoperability between separately composed unit services, the information required for composition must be provided between each unit system and automatic execution should be carried out between each service. As an example, before going on a business trip, reservations for plane tickets, hotel rooms and rental cars must be made separately but these unit services must at the same time be well organized to form a good composition. For such purposes the extended DAML-S capable of rule based searches is suggested.

(3) Service Quality Measurement
In order to provide reliable services, an automated semantic web services method is required. However, the DAML-S that currently supports this function does not include rule information, and so the user's restrictions are not applied accurately. In order to solve such problems with DAML-S, this study uses DamlRuleML [6], the DAML-OIL ontology language for RuleML, to include rule information in DAML-S and produce an extended DAML-S improved model. Since DamlRuleML is based on the grammar of DAML+OIL, the matter of compatibility with DAML-S specifications shall not be a problem here.

(4) Satisfaction with Search Results
The semantic management module suggested in a former study [2] is used for providing priority ranked search results. The ranking of service results is supported by using the newly suggested similarity measurement model [2].

4. Matchmaking Algorithm

4.1 The Definition of Matchmaking and Requests

Matchmaking is a process of finding the service provider that satisfies the server requester's requests. Matchmaking is executed based on whether the web service request and web service advertisement match or not. The match between requests and advertisements is determined based on whether the service input and output among the functional description match or not. The matchmaking system must support input and output through the repository and enable service browsing, correction and cancellation[12].

4.2 Matchmaking Algorithm

For efficient semantic web service searching, matching service requests and service advertisements must be done accurately. The match between requests and advertisements is made based on the match between inputs and outputs of the functional description. In other words, when the factors of the service request input and the service advertisement input match each other, the two inputs match, and when factors of the service request output and factors of the service advertisement output match each other, the two outputs match. As so, when all inputs and outputs match, the service executes the service request appropriately and provides satisfying results. In this research, whether the input and output match or not is judged by classifying the matches into five different levels: Exact, PlugIn, Subsume, Intersection and Fail[12].

[Rule 1] Exact
If advertisement A and request R are equivalent concepts, we call the match Exact. ($R = A$)

[Rule 2] PlugIn
If request R is super-concept of advertisement A, we call the match PlugIn. (R □ A)

[Rule 3] Subsume
If request R is sub-concept of advertisement A, we call the match Subsume. (R □ A)

[Rule 4] Intersection
If the intersection of advertisement A and request R is satisfiable, we call the match Intersection ($R \cap A$)

[Rule 5] Fail
If advertisement A and request R are not equivalent concepts, we call the match Fail ($R \neq A$)

4.3 Matchmaking Algorithm Application

4.3.1 Semantic Web Services Scenario
A customer wants to purchase a cellular phone through web service. The following describes the

processing results after web service request. Mr. Park, a car salesman, wants to purchase a camera built in mobile phone made by Samsung so he can show the newest car models to his customers. But if the Samsung phones cost over 200 thousand won he considers purchasing an OEM brand product instead. Mr. Park used the web service to buy a cellular phone that satisfies such conditions. Here, the extended semantic web services system's web agent obtained information from Mr. Park's personal information profile that he prefers Hanmac phones among OEM brand products and within the same cost range phones with more functions. Based on this information, Hanmac's camera built in cellular phone costing below 200 thousand won was recommended to Mr. Park. The web service even executes the purchase process if Mr. Park decides to buy the recommended product.

4.3.2 Definition of Ontology

This section suggests the definition of the ontology based on the scenario given in Section 4.3.1. In order to define the ontology, the advertisement and request services were defined first applying the DAML-S 0.9 version services profile class. In addition, the syntax rules were expressed using DL notions of the DAML+OIL syntax and the expression method is as shown below.

ServiceProfile □ □
Advertisement □ ServiceProfile
Query □ ServiceProfile
Sales □ (=l providedBy.Provider) ∩
(=l hasFeatureSelection.FeatureTyper) ∩
(=l hasQuantity.Integer) ∩
(=l hasProductPrice.Integer)
Provider □ (=l hasName.ProviderName) ∩
(=l hasCompanyName.CompanyName)
Product ≡ hasBuilt-inCamera.Item
Item ≡ Cellular-Phone U PDA

5 Intelligent Web Services System

Section 5 will suggest the methodology, technical factors, overall structure and functions of each module for developing an accurate and efficient web services searching system and in the end provide the final screen of the system realized.

5.1 The Background and Necessity for Research

Semantic Web Services, a combination of semantic web technologies and web services, is a new paradigm enabling information exchange through the Internet. Also being an application interface capable of interaction between two different types of applications, the development of an e-business framework using this is expected in the near future. However, current web services technologies merely provide services not reflecting the user's requests and no measurements are made of the web services quality. As so, this study is purported for enabling more reliable and efficient web services searching and organization by designing and realizing an improved web services searching system differentiated from the existing one. By applying the ranking algorithm to the suggested system more accurate and reliable results are presented.

5.2 Technical Factors of Intelligent Web Services

This section will deal with the technical factors and standards of the proposed system.

Table 1.Technical Points

Semantic Web		**Web Service**	
Technical factor	**Standard**	**Standard**	**Technical factor**
Rule based Service	**Extended DAML-S (DAML-S + DamlRuleML)**		Rule based Service
Ontology	DAML-OIL	**Extended UDDI**	Discovery
RDF Schema	RDF(S)	WSDL	Description
	RDF	SOAP	Wire
Security	XML Encryption, XML Digital Signature		Security
Syntax	XML, XSD		Syntax
Network Protocols	HTTP, FTP, SMTP		Network Protocols

5.3 System Architecture

In this section, we propose an overview of our suggested system.

Figure 3 depicts the system model's general flow chart indicating the applied methodologies. It shows the whole procedure from the user request query to the results provided to the user. The suggested system model executes web service searches along a 10-step procedure. The ten steps are described below.

Step 1: User's Request Query
The user makes a search query through the user interface.

Step 2: Request Search for Analogous Terms Groups
Search for groups of analogous terms is requested based on the returned search results. The E-engine ontology server provides the user with ontology terms and descriptions so that the user is able to select the right ontology.

Step 3: Production of Analogous Terms
A re-query is made in order to select the desired term among the search results of analogous terms groups. For example, food, groceries, cooking etc. are terms that would be placed in the group "foodstuffs" and terms such as original equipment manufacturing in the group "OEM." As so, the desired result can be drawn by the user's re-query.

Step 6: Request for Rule Based Search
Rule based search is requested based on the produced results.

Step 7: Execution of Rule Based Search
The results produced in step 4) and the user's rules acquired from the user profile registry are compared to select only the appropriate information.

Step 8: Request for Analysis of Extracted Results
For analysis of the extracted results, similarity measurement and ranking is done by requesting to the semantic management module, the classification processing module suggested in current studies.

Step 9,10: Return and Store Extracted Results
Even though the information of the service desired by the user is obtained through a rule based search, it is impossible to perceive whether the web service is currently valid and can be used by the user immediately. Due to this, the invalid services are extracted and ranked based on the user's search query through the similarity measurement model suggested in current studies [2]. Then the final results are returned and the information is stored in the registry.

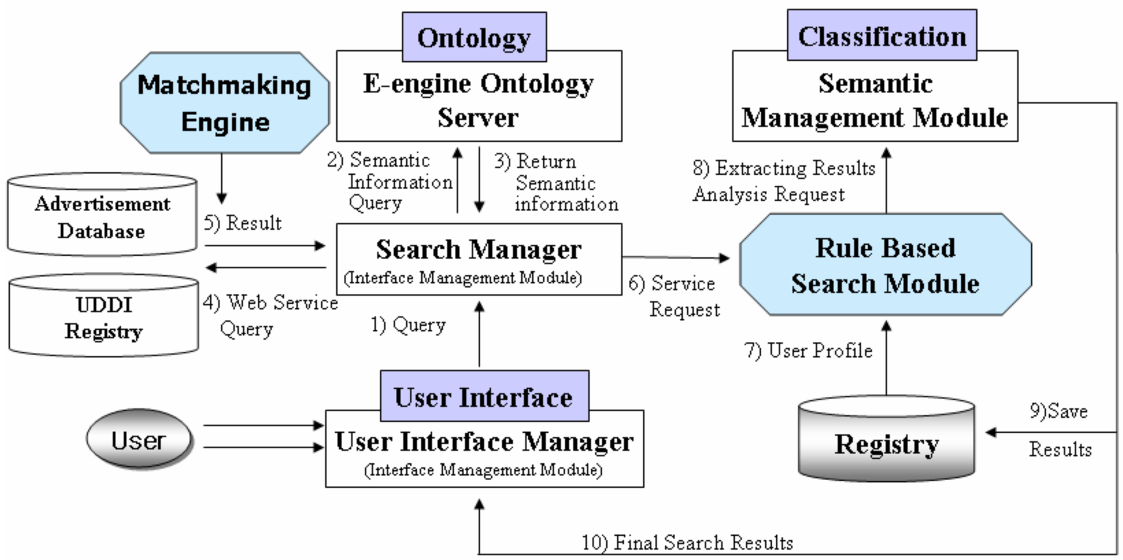

Fig. 3. System Architecture

Step 4 ,5: UDDI Search and Returning Search Results
The search for the user's search query is executed by using UDDI and the advertisement database (advertisement registry). Then the returned search results are analyzed using the matchmaking engine and the results are sent back.

6 Conclusions

Semantic Web Services is a technology combining the SOAP, WSDL and UDDI standards based web services and semantic web technologies, such as RDF,

DAML-OIL and OWL, to enable discovery, execution and compounding of web services.
However, the currently used web services searching system does not make any considerations of the user's requests, does not make measurements of quality such as reliability nor supports search result ranking.
To overcome the disadvantages of the current system, this paper proposed the extended Semantic Web Services searching system, which is a fusion of the semantic web technology and the web services technology. The proposed system supports UDDI extension to have users requests considered and verified and also supports search results ranking in order to provide more accurate and reliable service.

Acknowledgements

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2006-353-B00059).

References:

[1] M. Nodine, W. Bohrer, and A. Ngu. Semantic multibrokering over dynamic heterogeneous data sources in infosleuth. In Proc. of the International Conference on Data Engineering, 1999.

[2] Okkyung Choi, Seokhyun Yoon, Myeongeun Oh. Sanygoung Han, "Semantic web Search Model for information retrieval of the semantic data", The Second HSI Conference, June, 2003, pp. 588-593.

[3] RDQL(A Query Language for RDF), http://www.w3.org/Submission/2004/SUBM-RDQL-20040109

[4] Okkyung Choi, Sangyong Han, Ajith Abraham, "Extended Semantic Web Services Model for Automatic Integrated Framework", International Conference on Next Generation Web Services Practices (NWeSP'05), 2005.8.

[5] Sycara K., Paolucci M., Soudry J., Srinivasan, N.: Dynamic Discovery and Coordination of Agent-Based Semantic Web Services, IEEE Internet Computing, Vol. 08, no. 3, pp.66-73, May/Jun, 2004.

[6] DamlRuleML, http://userpages.umbc.edu/~mgandh1/DamlRuleML.html

[7] A. Ankolekar, M. Burstein, J. R. Hobbs, O. Lassila, D. L. Martin, S. A. McIlraith, S. Narayanan, M. Paolucci, T. Payne, K. Sycara, and H. Zeng. "Daml-s: Semantic markup for web services" In Proc. of the International Semantic Web Workshop, 2001.

[8] Massimo Paolucci, Julien Soudry, Naveen Srinivasan and Katia Sycara,"A Broker for OWL-S Web services", The Robotics Institute, Carnegie Mellon University.

[9] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara, Semantic matching of web services capabilities. In Proc. Of the 1st International Semantic Web Conference (ISWC), 2002.

[10] Andy Seaborne, "Jena Tutorial: A Programmer's Introduction to RDQL", http://jena.sourceforge.net/tutorial/RDQL/#RDQL-by-Example, April, 2002.

[11] Garofalakis, J., Panagis, Y., Sakkopoulos, E., Tsakalidis, A.: "Web Service Discovery Mechanisms: Looking for a Needle in a Haystack?", International Workshop on Web Engineering, 2004.

[12] Okkyung Choi, Sangyong Han and Ajith Abraham, Semantic Matchmaking Services Model for the intelligent Web Services, International Conference on Computational Science and Applications, UK, IEE Press, UK, pp. 146-148, 2006

[13] LDL++ Java Interface, http://pike.cs.ucla.edu/ldl/