# Semantic Matchmaking Services Model for the intelligent Web Services

Okkyung Choi, Sangyong Han, Ajith Abraham

Department of Computer Science & Engineering, Chungang University,221, Huksuk-dong, Dongjak-ku, Seoul, 156-756, Korea
okchoi@archi.cse.cau.ac.kr, hansy@cau.ac.kr, ajith.abraham@ieee.org

**Abstract.** Semantic Web Services enable discovery, execution and composition of automated web services by combining web services based on standards, such as SOAP, WSDL and UDDI, with semantic web technologies such as RDF, DAML+OIL and OWL. In this paper, Matchmaking Services Model is suggested for the intelligent Web Services. The suggested model enables efficient matches between service requestors and service providers with the matchmaking algorithm.

## 1. Introduction

Web services are one of the key technologies in e-business and presently research and development of languages for constructing semantic web services, such as DAML-S, WSPL, X-LANG and BPEL4WS, are underway in various fields. As for DAML-S, a method for accessing the existing web services method from the semantic web environment, weak points of former methods have been improved to enable effective web services registration, search, organization, execution and composition. However, the current semantic web services model has some disadvantages in supporting automated web services. For such reasons, this paper suggests the Semantic Matchmaking Service Model to solve the above problems and enable efficient web services search and construction.

## 2. Matchmaking Services Model

### 2.1 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.

### 2.2 Matchmaking Algorithm

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.

[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. ($R \supset A$)
[Rule 3] Subsume
If request R is sub-concept of advertisement A,
we call the match Subsume. ($R \subset A$)
[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 ($R \neq A$)

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. As the level goes up from [Rule 1] to [Rule 5], the ranking is lower.
The match ranking method applied in this research is largely divided into Steps 1 and 2. In Step 2, a new ranking algorithm [4], a modification of the former vector model, is applied for [Rule 2] and [Rule 3] to produce more detailed ranking. This newly suggested match ranking algorithm is described as follows:

**Step 1**

```
First_Match_Compare(output.R, output.A) {
   if output.R is equivalent to output.A then Level = Exact; return Exact;
   else if output.R is SuperClassOf output.A then Level = PlugIn; return PlugIn;
   else if output.R is SubClassOf output.A then Level = Subsume; return Subsume;
   else if output.R is not incompatable with output.A then Level = InterSection; return
   InterSection;
   else Level= Fail; return Fail;
}
R:request, A:Advertisement
```

**Step 2**

```
Secound_Match_Compare(output.R, output.A) {
         Switch(Level) {
         case Exact:
                  Level_rank = 0;
```

```
            Break;
    case PlugIn: case Subsume:
            Call Function Ranking_Compare();
            Break;
    }
}
```

The two-step match ranking algorithm is applied in the case where the matching levels are Exact, Plugin and Subsume. When the matching level is Exact, it means that the service request and the service advertisement are exactly the same and so this level is ranked at the highest match rank. In the case where the service request comprises the service advertisement, the Ranking_Compare() function dealt with in the former study Semantic Management Model [4] is called. The Ranking_Compare() function is indispensable to ranking the services within the same level. It uses the relationship, that is, the vertical and horizontal closeness, between succeeding levels and the synonym relation between terms to rank the matches. As so, a more detailed –two-step match ranking method is produced to improve the former one-step simple match ranking method in order to provide clearer priority ranking of search results and more accurate and efficient search results.

## 3. Conclusion

This paper suggests the Semantic Matchmaking Service Model. For efficient semantic web service searching, matching service requests and service advertisements must be done accurately. The suggested model allows verification and also supports search results ranking in order to provide more accurate and reliable service.

## Acknowledgements

This research was supported by the MIC (Ministry of Information and Communication), Korea, under the Chung-Ang University HNRC-ITRC (Home Network Research Center) support program supervised by the IITA-Institute of Information Technology Assessment.

## References

1. Daniel A.Menasce, "QoS Issues in Web Services", IEEE Internet Computing, 2002.12
2. M. Nodine, W. Bohrer, and A. Ngu. Semantic Brokering over dynamic heterogeneous data sources in infosleuth. In Proc. of the International Conference on Data Engineering, 1999.
3. David Trastour, Claudio Bartolini and Javier Gonzalez-Castillo, "A Semantic Web Approach to Service Description for Matchmaking of Services", HP Labs Technical Report.
4. Okkyung Choi, Sangyong Han, "SW-IQS: Semantic Web based Information Query System for the integration of semantic data", Presentation of Studies at the Korea Information Processing Society Fall Edition, 2004