Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 10-11, 2007

Development of an Web Service Architecture for Enterprise Application Integration

Ji-Hyeon Kim, Jae-Cheon Jung, Young-Woo Chang, Hoon-Seon Chang, Jae-Cheol Kim, Hang-Bae Kim
Korea Power Engineering Company (KOPEC)
jhkim10@kopec.co.kr
Kyu-Ho Kim, Dong-Chul Lee
Korea Electric Power Data Network (KDN)
khkim@kdn.com

1. Introduction

The purpose of Enterprise Application Integration (EAI) is to enable the interoperability between two or more enterprise software systems. These systems, for example, can be an Enterprise Resource Planning (ERP) system, an Enterprise Asset Management (EAM) system or a Condition Monitoring system. Traditional EAI approach, based on point-to-point connection, is expensive, vendor specific with limited modules and restricted interoperability with other ERPs and applications. To overcome these drawbacks, the Web Service based EAI has emerged. It allows the integration without point to point linking and with less costs. Many approaches of Web service based EAI are combined with ORACLE, SAP, PeopleSoft, WebSphere, SIEBEL etc. as a system integration platform. The approach still has the restriction that only predefined clients can access the services. This means clients must know exactly the protocol for calling the services and if they don't have the access information they never can get the services. This is because these Web services are based on syntactic service description. In this paper, a semantic based EAI approach, that allows the uninformed clients to access the services, is introduced. The semantic EAI is designed with the Web services that have semantic service descriptions. The Semantic Web Services(SWS) are described in Web Ontology Language for Services(OWL-S)[1], a semantic service ontology language, and advertised in Universal Description, Discovery and Integration (UDDI). Clients find desired services through the UDDI and get services from service providers through Web Service Description Language(WSDL).

2. Enterprise Application Integration Architecture

Figure 1 shows the Service Oriented Architecture (SOA) of Enterprise Applications. The Enterprise Application Integration is implemented through Web service broker hub. The SWS Interface in Figure 1 plays the role of the broker hub. It receives service requests from clients, locates and selects the appropriate services, interacts with service providers and receives the service results and returns the results to clients[4]. The proposed SOA used open industry-standard specification and does not have vendor specific platform tools for integration.

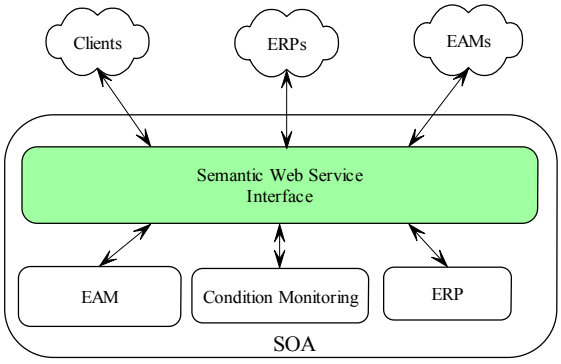

Figure 1. SOA of Enterprise Application

The proposed SOA is different from traditional integration approach in that it has only one service interface instead of one-to-one mapping service interface for each service provider.

3. Semantic Web Service Interface

Figure 2 shows the SWS Interface internal architecture that consists of Interface Agent(IA), Web service ontology, and Concepts and Terminology ontology. The Interface Agent plays a key role of finding services from advertised services, invoking appropriate services and returning the serviced results. The service ontology described in OWL-S is for service finding and grounding. The service grounding[2] is the connection process between the semantic interface description and WSDL. The Concepts and Terminology ontology is used to enhance semantic meaning of terms that are used in service findings.

3.1.OWL-S/UDDI Service description

Web Ontology Language(OWL) facilitates greater machine interpretability of Web contents than that supported by XML, RDF, and RDF Schema (RDF-S), by providing additional vocabulary along with a formal semantics. OWL-S is an application of OWL that describes the ontology in Web service specific point of view. Services of EAM are advertised in Web in the OWL-S format and then clients can find and invoke the services. OWL-S is composed of four category of

Service, Profile, Process and Grounding ontology. Among them the Profile ontology is used in service finding in suggested SWS Interface. The input, output, precondition and effect of Profile ontology can be mapped into UDDI Business Service model[3] and can be advertised through UDDI. The service advertisement of UDDI has advantages because UDDI supports Internet-wide network of registries of Web services for businesses to quickly, easily, and dynamically discover Web services.

3.2. Interface Agent

Once the Interface Agent(IA) receives the service request from clients, it finds the appropriate service from providers. This service location process includes service matchmaking with advertised services in OWL-S/UDDI. The matchmaking is the process of finding services and there are predefined matching types[5] according to the degree of match[3]. The IA selects the best services among the located services by referring the matched types and then gets the services from the selected service providers by WSDL grounding. The service finding is based on input, precondition, output, effect and the service results will be returned as output and effect.

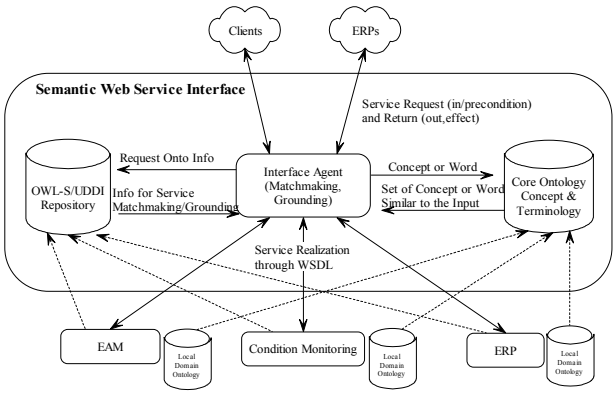

Figure 2. Internal Architecture for Semantic Web Service Interface

3.3. Concepts and Terminology Ontology

Concept is the class for the objects, and Terminology is the primitive vocabulary. The input, precondition, output, effect are falling into these two categories. The Concept and Terminology ontology is defined in object modeling technique that the IA can refer the ontology in matchmaking process. This mechanism can enrich the semantic meanings of terms used in SWS Interface. For example, the term Gas Insulation Switch(GIS), one of a substation equipment that includes Gas Circuit Breaker(GCB)s and Disconnecting Switch(DS)s, can be used instead of GCB if there is no Web services supporting the GCB[6].

4. Conclusion

The semantic Enterprise Application Integration using Semantic Web Service is suggested. The SWS is based on the OWL-S service description ontology. The Concept and Terminology ontology is also used to increase vocabulary semantics. The Interface Agent is the core component in the SRS Interface that performs service finding, invoking and returning the service results to clients. The proposed mechanism increases Enterprise Application system independence and modularity, and ensure that they can be easily plugged in to support the business process.

REFERENCES

[1] Martin, D , Burstein, M, Hobbs, J, Lassila, O,McDermott, D, McIlraith, S, Narayanan, S, Paolucci, P,Parsia, B, Payne, T, Sirin, E, Srinivasan, N ,Sycara, K ,"OWL-S Semantic Markup for Web Services", W3C submission 2004, http://www.w3.org/Submission/OWL-S/

[2] Jacek Kopecký, Dumitru Roman, Matthew Moran, Dieter Fensel , "Semantic Web Service Grounding", Proceedings of the International Conference on Internet and Web Applications and Services (ICIW'06), February 2006.

[3] Srinivasan, N, Paolucci, M and Sycara, K, "Adding OWL-S to UDDI, implementation and throughput", First International Workshop on Semantic Web Services and Web Process Composition, 2004.

[4] M. Paolucci, K. Sycara, and T. Kawamura, Delivering Semantic Web Services, tech. report CMU-RI-TR-02-32, Robotics Institute, Carnegie Mellon University, May, 2003

[5] J.-H. Kim, W. Kim. Y.-J. Kim & J.-H. Kang: Extended Planning Mechanism to Increase Web Service Utilization, Proc. APWEB 2005 Conference, Shanghai, China.

[6] Ji-Hyeon Kim et al, Case Based Asset Maintenance for the Electric Equipments, Transaction of the Korean Nuclear Society Spring Meeting, May 2006.