# A Study on Design of Web Service System using UDDI Registry

Tae Hee Kim[a] and Sun Kyung Kim[b]

[a] Computer and Information Engineering, Daegu University
15, Naeri-ri, Jinryang-eub, Gyeongsan-si, Gyeongbuk, 712-714, Korea

[b] Computer and Information Technology Engineerin., Daegu University
15, Naeri-ri, Jinryang-eub, Gyeongsan-si, Gyeongbuk, 712-714, Korea
Tel: +82- 53-850-6582, Fax: +82- 53-850-6589, E-mail:skkim@daegu.ac.kr

**Abstract:** *Recent web environment needs a web service method that can integrate the individual web service. The UDDI registry can accommodate this need by classifying the information of the web service system for an easy retrieval of the information, thus it reduces the time to find a specific web service. Using the basic information search function of the UDDI registry, the web provider can offer various types of information to the client. In this paper, we implemented the process of registering the information to the UDDI registry and retrieving the various types of information from the UDDI registry in a given format.*

**Keywords:** *Web service; Distributed system; UDDI*

## Introduction

There are various types of protocols in INTERNET. These protocols are connection-oriented type so that they have the problems of wasting resources and brittle security. Thus they need new computing technical skill based on the distributed component because the web environment is so complex itself. As a result, the web service skill was born to unify all different protocols. Web service system is the standard model of a server application program in distributed environment, which is a set of remote procedures that can be called by remote client. Web service is defined as the interface that describes the accessible operator in XML message[1], and it makes on-line business possible by using standardized software skill that connects the system programs of different platforms. Universal Description, Discovery and Integration (UDDI) is the standarization of storing meta data and their retrieving method for the web service. It can efficiently search the web service and retrieve the data. UDDI registry has the sorted information about web service. If the well sorted information is present, it will save time needed for random retrieval. For example, one can find a certain company if information on companies are well sorted by regions or types of business. If copy facility exists between UDDI registries, it can be used more generally. Many company propel UDDI project. UDDI defines unified rules about information expression and retrieval on web.

In this paper, we studied storing XML data in relational database when provider puts up information on UDDI, and implemented web service system that can offer the retrieval service to the user.

# Related Works

## The schema of UDDI

UDDI is the data storage based on XML. It can work independently of the client environment by using SOAP message and can exchange data among UDDI registries. UDDI API element defines the XML element for creation, update, and deletion of the data, and it consists of Publishing API element for creating, modifying, and deleting information and Inquiry API element for finding business information in UDDI.

The function of UDDI is similar to that of database system but the details are different. To store information and access it in database system, JDBC driver must be installed on the client system since JDBC driver uses different protocol. To access UDDI registry, client systems use SOAP message, so they do not need to install the provider. To retrieve and put on business information on UDDI, it needs to create a common object for Java API for XML Registries (JAXR) clients. These are the order to create object.

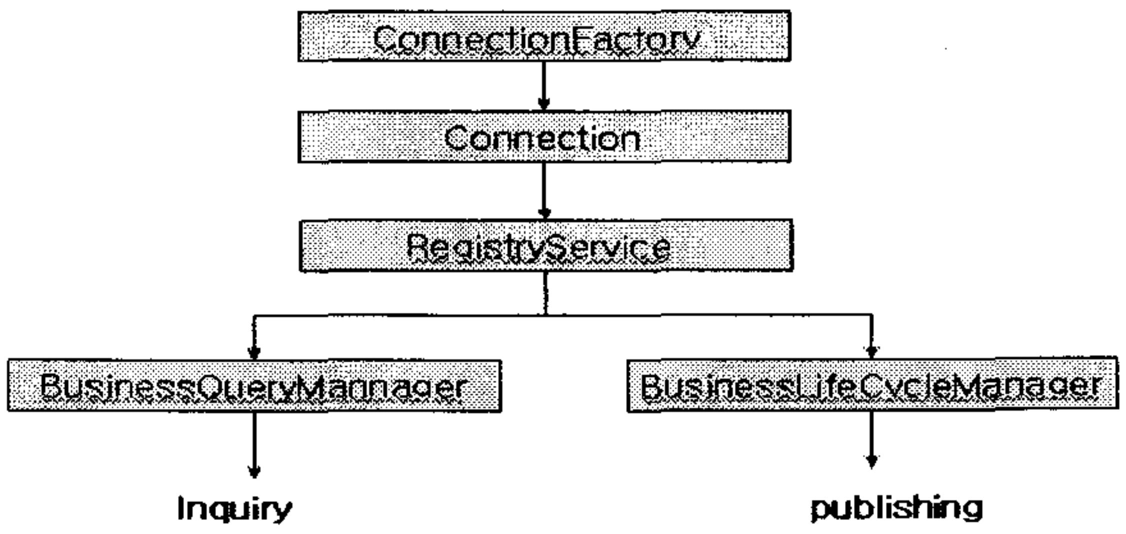


**Figure 1 - The order of creating object**

. ConnectionFactory object is a factory object to make Connection object.
. Connection object is the object to connect JAXR client with UDDI registry.
. RegistryService object provides a method to obtain BusinessQueryManager object and BusinesslifeCycleManager through JAXR provider.
. BusinessQueryManager object is used when JAXR client inquires the bussiness information on UDDI registry.
. BusinessLifeCycleManager object is used when client publishes the bussiness information on UDDI registry.

UDDI registry data architecture is constructed to represent the service and business information.[2] These are the business information elements for registry and retrieval.

. businessEntity
. businessService
. buindingTemplate
. tModel

tModel uses meta data technique that can be referenced by every object in UDDI registry. UDDI registry contains tModel definition and user can additionally register the tModel element to UDDI registry. The tModel element also offers name space.

tModel element describes URL path of WSDL document which defines the class of method and the data type for the web service.

### Map the XML document to the relational schema

The document on the web is expressed with XML, a standard data-oriented language. XML document is represented as tree-type graph and schema can be extracted from it. Each element is represented as node and each node has its line. Each line between the nodes has identification name and its own direction[3]. In the database view point, XML is rather suitable for expressing the structure or the information of document than expressing contents. To use the relational database (RDB) system of strict schema structure, mapping of the XML structure into RDB is needed. To store information of XML structure into relational database table, it is necessary to consider the difference between the hierarchical XML structure and the planar database structure. There are two methods, model mapping and structure mapping[4], to store XML document into relational database. The structure mapping method first analyzes the elements that represent structure information, then defines relational schema and stores XML document. For this, it is necessary to convert a schema into DTD independent type and assign independent ID with location information of nodes to the tables to have the capability of adding and deleting element for efficient retrieval. We can use fragmentation method or virtual fragmentation method to save the structure information of XML document into the relational database. Fragmentation method separates XML document into unit element in order to save the structure information and during retrieval it searches the corresponding element or combination of the lower-level elements. In case of adding, modifying, and deleting, only the related element needs to be modified, so this method is easy to edit and manage. However, organizing the combination of elements needs longer retrieving time. In addition one should reconstruct the related database if DTD structure information changes. Virtual fragmentation method saves the start offset and the end offset of the document into the relational database system. Since this method does not require combining process, structure search is possible by using location information during node retrieval, thus searching efficiency of virtual fragmentation method is higher than decomposition method. In spite of this merit, location information of other elements should be updated if the element is added, deleted, or modified. Due to this overhead and difficulty in keeping database consistency, this system is suitable for the system doing mostly searching.

## Implementation of web service System

For system environment, we used JWSDP-based web logic server on window XP operating system and MS-SQL Server. JAVA was used for implementation language and IBM test registry was used for UDDI registry .

### System structure

The structure of the system implemented on JAX-RPC based JWSDP is given in Fig. 2 and the role of each components is as following.
1) EJB

Performs logic to store and retrieve information to and from database. JDBC is used for database.

2) Web service system and WSDL

Calls the corresponding EJB remote method after receiving SOAP message asking retrieval request, then sends result consisted of SOAP message to client.

3) UDDI registry

Put up the system information of web service to help client to use.

4) JSP

Client calls remote procedure that can acquire information offered by business companies.

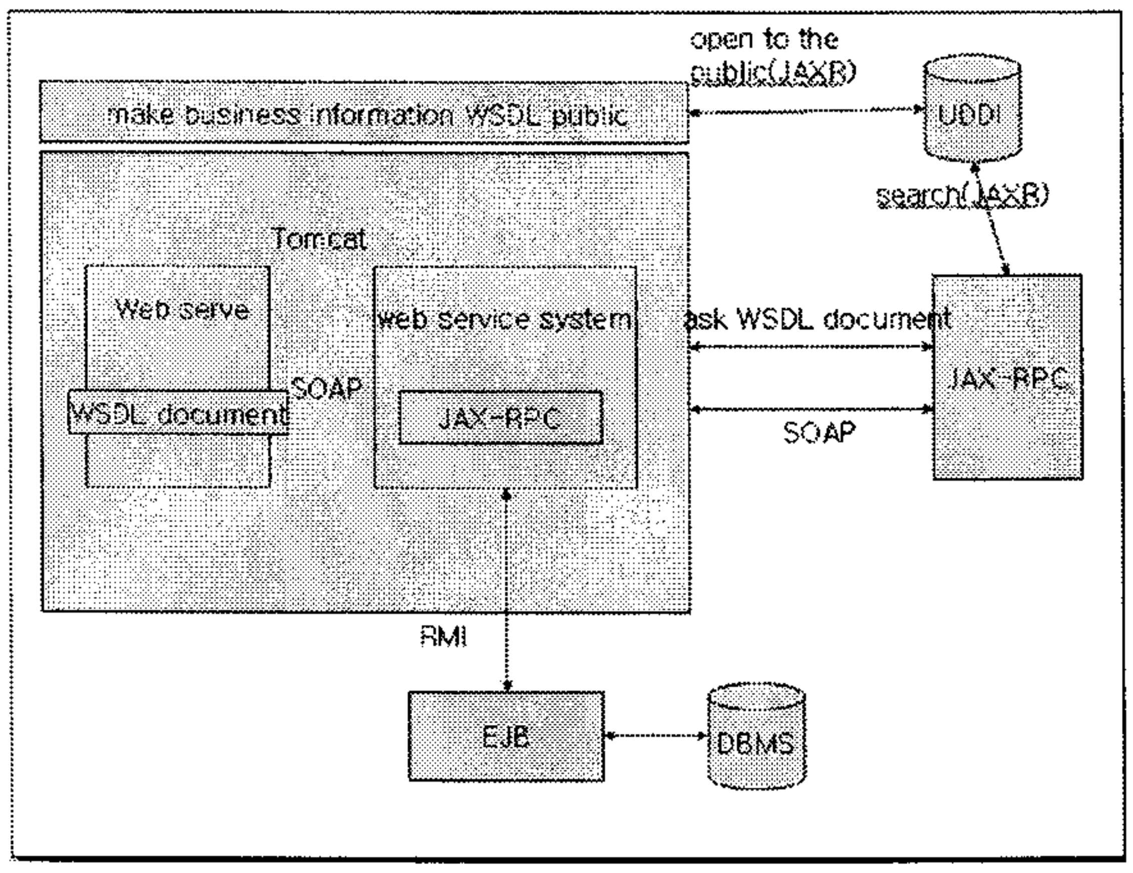
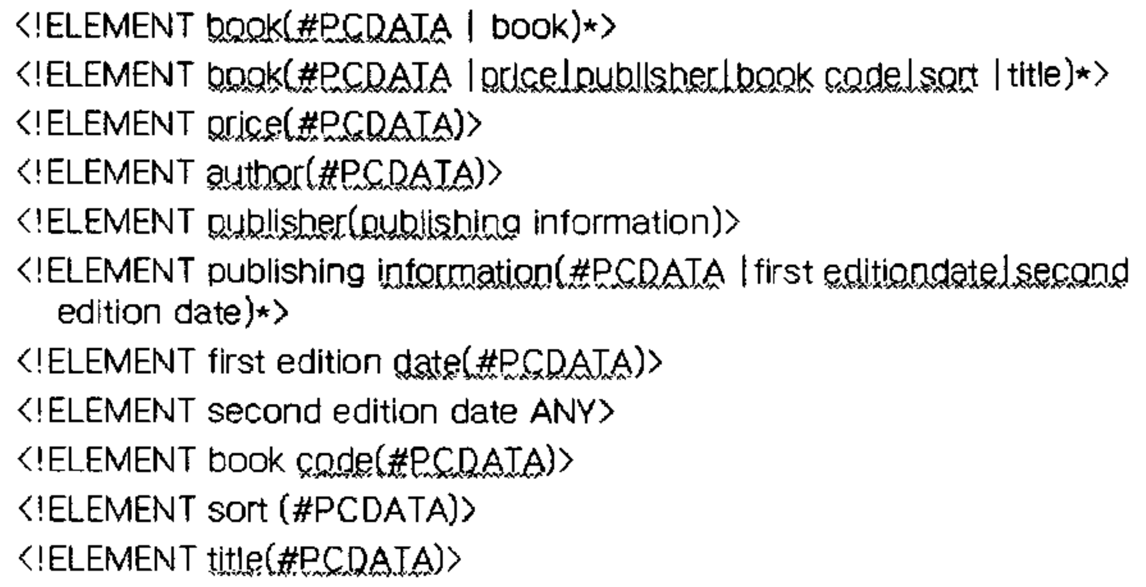


**Figure 2 - Structure of web service system**

## The process of XML data

To implement the process in the relational database, extract the schema for processing inquiry to XML and then achieve element and structure information contained in DTD.



```
<!ELEMENT book(#PCDATA | book)*>
<!ELEMENT book(#PCDATA |price|publisher|book code|sort |title)*>
<!ELEMENT price(#PCDATA)>
<!ELEMENT author(#PCDATA)>
<!ELEMENT publisher(publishing information)>
<!ELEMENT publishing information(#PCDATA |first editiondate|second
   edition date)*>
<!ELEMENT first edition date(#PCDATA)>
<!ELEMENT second edition date ANY>
<!ELEMENT book code(#PCDATA)>
<!ELEMENT sort (#PCDATA)>
<!ELEMENT title(#PCDATA)>
```

**Figure 3 -The structure of element**

For structure mapping, we analyze elements that contain structure information and define relational schema. To create relational database table, analyze DTD and structure route information table. Express XML document in binary tree that structured child at left, brother at right. Give a unique ID to each element and map it to entry.

Tables are needed to store XML document, to express detailed information of the XML document, general information of full document that contains pathID to express route information, pathID to distinguish each path, sID to express to start ID, eID to express to end ID, parID to parent ID, and list of all the elements. After creating tables, implementation was performed in the following sequence.

1) Construct the remote interface to define remote procedure

2) Construct the class to implement the remote interface

3) Construct DD of disposition descriptor.

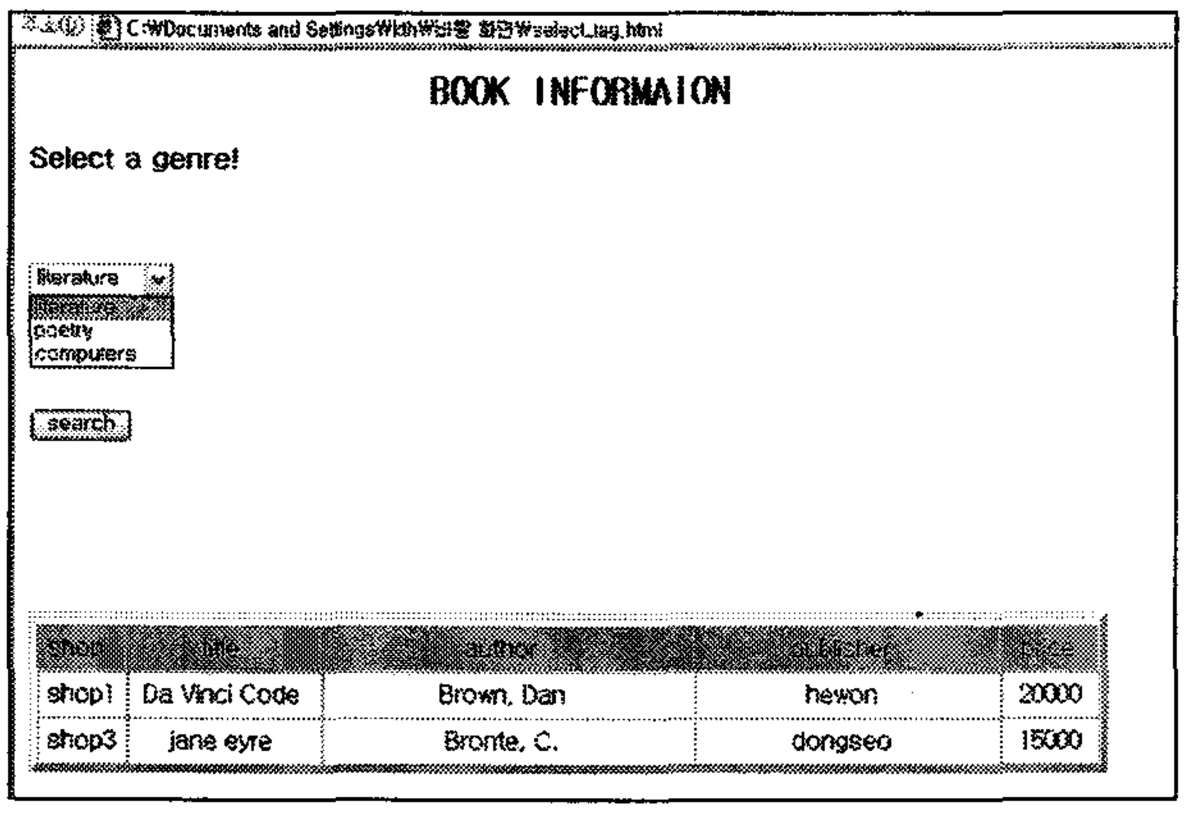
Construct disposition descriptor web.xml that has the contents to be put in servelet container. The contents have the information about session, URL mapping, initialization, default document which must be present even it does not have contents to apply. And construct RML interface disposition descriptor, jaxrpc-ri.xml, which automatically creates tie class and WSDL document.

4) Create tie class and WSDL document

Tie class and WSDL document are automatically created for packaging and disposition and then added to the WAR file.

5) WAR and disposition

After making WAR, the final product of web service system, dispose it to the servelet container to do the remote procedure call by client.



BOOK INFORMAION

Select a genre!

| shop | title | author | publisher | price |
|---|---|---|---|---|
| shop1 | Da Vinci Code | Brown, Dan | hewon | 20000 |
| shop3 | jane eyre | Bronte, C. | dongseo | 15000 |

**Figure 4 - Result screen**

## Conclusions

In this paper, we implemented registering and retrieving information to and from UDDI using client side application program, JAVA. It can put information of various types and makes possible for clients to get the information of what they want. UDDI 3.0 is the important skill of web service, but UDDI is in development stage yet because of complicated standard. We investigated the possibility of web service that applies UDDI registry. A study on various types of data process and how to achieve higher speed on retrieval are future research topic.

## References [blank line]:

[1] Kreger, H. (2001). *Web Services Conceptual Architecture* (WSCA 1.0), IBM Software Group.

[2] *UDDI version 3.0 Published Specification*, http://uddi.org/pubs/uddi-v3.00-published-20020719

[3] Goldman, R., and Widom, J. (1997). “Data Guide : Enabling Query Formulation and Optimization in Semistructured Databases”, *In Proceeding of VLDB.*
[4] Florescu, P., and Kossman, D. (1999). "Storaging and Querying XML Data Using an RDBMS", *IEEE Data Engineering Bulletin*, vol. 22, pp.27-34.
[5] Malaika, S. (1999). "Using XML in Relational Database”, *15th Conf. on Data Engeneering*, Australia, p.167.