

## **Comparative Analysis of UDDI and ebXML Registry in Service Oriented Architecture**

**Dr. Netra Patil**

Sinhgad Institute of Business Administration and Research

netra.patil@sinhgad.edu

**Abstract** – The automation of business processes across organizations required to integrate the cooperating software systems in order to achieve the desired level of interoperability. Modern integration technologies are used across business organizations to perform business electronically. In this paper, we have discussed and compared significant features of the two such technologies UDDI (Universal Description, Discovery, and Integration) and ebXML (electronic business XML) which serve as a registry for publishing and finding a web service for various business organizations. After introducing the concept of both the technologies, similarities and differences between them are discussed in this paper. Business organizations wanting to do business-to-business collaboration need to decide, which technology from these two will be suitable for their collaboration requirement. This paper gives a fair comparison on the architectural, technical, and vendor support basis for these two technologies.

**Keywords** - UDDI; ebXML, EDI, B2B, UN/CEFACT, CPP, CPA

### **I. INTRODUCTION**

In early years, for automating business processes across organizations, electronic mail was used to communicate between people, and web applications were used to communicate between human and business applications published on internet. Still business processes can not be automated completely because of high implementation cost. The main reason is that business applications of cooperating business organizations must be developed in synchronous and should be interoperable. And that requires explicit requirements and constraints specification on the business application's interfaces. Such specifications are referred to as B2B protocols [1], or business protocols [5]. To exchange business documents between applications EDI systems were also used in 1960s which was not much standardized system earlier. In 1990 UN/EDIFACT provided an international standard for doing electronic business which was expensive and not widely used. EDI implementation is based on many different protocols and requires great skills which are very difficult to adopt for smaller companies. E-business flourished rapidly due to emerging internet and development of technologies like XML.

The two standards are emerging that could have very well impact on the way of conducting e-business in future. These are UDDI (Universal Description, Discovery and Integration) and ebXML (electronic business XML). UDDI is a vendor-sponsored initiative led by IBM, Microsoft, and Ariba,

whereas, ebXML is a UN/CEFACT (United Nations center for Trade Facilitation and Electronic Business) / OASIS sponsored initiative for creating a single global electronic market. UDDI and ebXML, make it possible for business organizations to publish information on the Internet about their products and web services, where the information can be readily and globally accessed by clients who want to do business

## II. E-BUSINESS USING UDDI

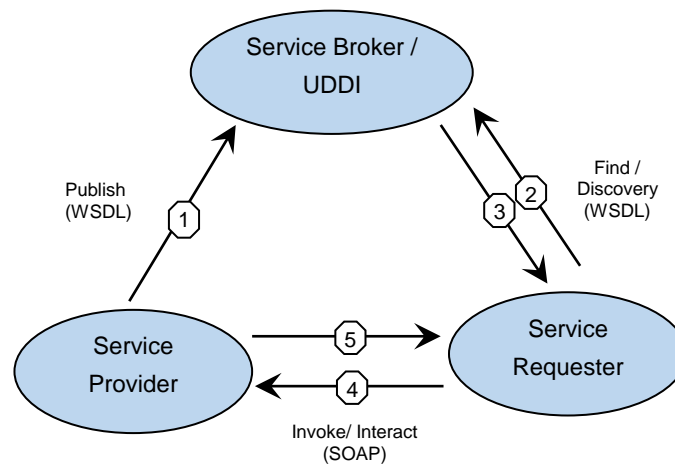


Fig. 1. Role of UDDI Registry in SOA

UDDI Registry is a web-based registry that exposes information about a business providing web service, web service and its technical interfaces. A service provider makes its services available to public users by publishing information about the service in a UDDI registry.

The information about Web services in a UDDI registry includes a description of the business and organizations that provide the services, a description of a service's business function, and a description of the technical interfaces to access and manage those services [9].

A UDDI registry consists of instances of four core data structures including the *businessEntity*, the *businessService*, the *bindingTemplate* and the *tModel*.

This information comprises everything a user needs to know to use a particular Web service. The *businessService* is a description of a service's business function, *businessEntity* describes the information about the organization that published the service, *bindingTemplate* describes the service's technical details, including a reference to the service's programmatic interface or API, and *tModel* defines various other attributes or metadata such as taxonomy and digital signatures [10].

### III. E-BUSINESS USING EBXML

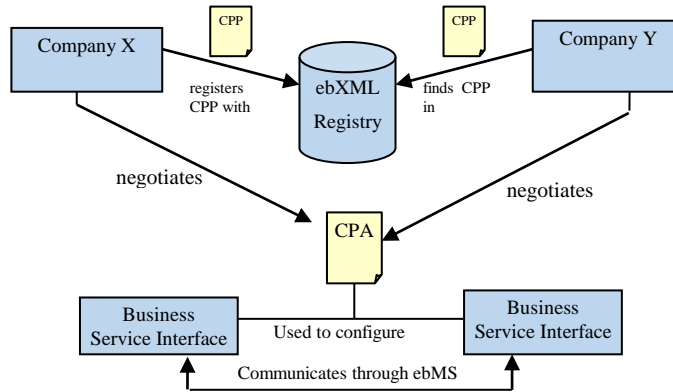


Fig. 2. Role of ebXML Registry

ebXML is a suite of specifications sponsored by OASIS and UN/CEFACT that enables organizations to conduct business over the Internet [11]. It provides an entire architecture and defines a new way of thinking about business and documenting it. With ebXML, companies are able to define how to conduct business using a specific vocabulary. ebXML registry contains descriptions of business artifacts and the repositories actually store them. Those artifacts are usually *Technical Architecture*, *Business Process Registry and Repository*, *Collaboration Protocol Profile and Agreement (CPP and CPA)*, *Message Services and Core components*.

With ebXML, companies are able to define how to conduct business using a specific vocabulary. *Core components* are used to build predefined documents. Messages are sent using standardized protocols and formats. All of this information is stored in ebXML registries. Business Processes and Business Document has to be created prior to their use. Specification of these both describes the workflow of business processes and the information exchanged between the partners respectively. These documents can be composed of reusable and extendable *Core Components*.

An ebXML Registry provides means for finding organizations, business processes, core components and other objects. Therefore it does not store the actual objects but metadata and associations between them. Business partners register their services in an ebXML registry along with their Collaboration Protocol Profiles (CPPs). During the search the registry is queried for a business partner that offers the required service. Based on the CPPs of both partners a Collaboration Protocol Agreement (CPA) is formed which specifies what kind of business is to be performed and how. Usually CPA is negotiated after being proposed by one party. Based on the agreement it is now possible to configure an ebXML enabled application and execute the business process.

### IV. COMPARATIVE ANALYSIS OF UDDI AND EBXML REGISTRY

UDDI and ebXML are built on the same technology but with different purpose. Both are based on Service Oriented Architecture, XML and are using SOAP as a message passing protocol, rely on registries to make discovery using dynamic binding possible. Both UDDI and ebXML put their services on a network and have means for service description, service discovery and service

invocation. However, the purpose of UDDI and ebXML differ: UDDI expose web services on the web so that anybody can call them from anywhere – a service-based approach; ebXML is intended for business partners to exchange well defined documents using an agreed business process – a contract-based approach. Compared to UDDI, the ebXML Repositories are intended for more general-purpose storage. UDDI is more specialized and geared towards the type of information that can be stored in the White, Yellow and Green pages. While UDDI stores mostly flat lists, ebXML is capable of handling classification information and information about relationships between business artifacts. In UDDI, usually, a web service is described, provided and published by one single entity (Unilateral). This service is then found and bound by the requester. Whereas, in ebXML approach is bilateral : two trading partners negotiate their profiles and come to an agreement based on which they execute their common business process. Compared to UDDI, ebXML is more at the executive level. UDDI follows bottom-up approach where specifications for specific core requirements are implemented and afterwards being assembled to build an integrated e-business suite. In contrast to that, ebXML follows a top-down approach in which, first the requirements for e-business are evaluated, and then the specifications are implemented to fulfill them. The bottom up approach increase the flexibility but all the requirements for e-business may not be fulfilled. ebXML is attempting to create a global market where enterprises of any size, anywhere can find each other and conduct business electronically using commercial off-the-shelf software. While the business partners interested in using UDDI registry are mainly from the IT industry only. In contrast to UDDI registry that offers only artifacts for discovery, ebXML registry also stores documents required for collaboration (CPP, CPA etc.) As both the registries focus on different aspects, their information models and architectures vary [6].

UDDI is not trying to own the world of e-business, but simply trying to facilitate all web-based services for query and introspection. By doing this with the collection off-the-shelf software to support it, the industry gets a worldwide repository of web-based services by default. On the other hand, there is a need for creating a global e-business standard, such as ebXML. However, creating a standard of this size and magnitude takes time and patience, which the industry either can't afford or chooses not to provide at this time. As a result, UDDI is clearly ahead at this time, but ebXML is favored to win.

UDDI specifications are well accepted and supported by industrial companies and W3C. Many large companies, such as IBM, Microsoft and SAP have their implementation of UDDI registries. Compared to UDDI, ebXML is less accepted. UN/CEFACT TMG estimates the acceptance rate of ebXML is only 3% of that of UDDI. Especially ebXML is less accepted by small and medium enterprises. Some of the companies having their ebXML registries are DISA, Seeburger, Drummond Group.

## V. CONCLUSION

In this study, we have studied two registries playing important role in e-business applications based on Service Oriented Architecture: UDDI and ebXML Registry. UDDI and ebXML have many things in common and can complement each other. In many aspects UDDI is architecturally superior to ebXML. The ebXML repositories are intended for more general purpose storage as compared to

UDDI registry. UDDI is more focused on the kind of information that can be stored in the White, Yellow and Green pages. The UDDI concentrate on middleware connectivity and describes the services companies use through XML, whereas, ebXML standardizes the way of using XML in B2B integration. UDDI contains mostly flat information while, ebXML store classification information and information about relationship between business artifacts.

One of these new technologies, such as UDDI and ebXML will definitely take strong position in upcoming e-business market. Many companies are still waiting to see which technology will be widely accepted before making investment in it. However, the standardization organizations try to increase operability of their approaches and providers try to integrate the new technologies in existing solutions.

## REFERENCES

1. C. Bussler. [2001] B2B Protocol Standards and their Role in Semantic B2B Integration Engines. In Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, volume 24, pages 3–11.
2. C. Bussler. [2001] B2B Protocol Standards and their Role in Semantic B2B Integration Engines. In Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, volume 24, pages 3–11. IEEE, 2001.
3. Nicholas Chase. [2002] Introduction to ebXML. <http://www-128.ibm.com/developerworks/edu/x-dw-xebxml-i.html>, 2002.
4. Birgit Hofreiter, Christian Huemer, and Wolfgang Klas. [2002] ebXML: Status, Research Issues, and Obstacles. In RIDE '02: Proceedings of the 12th International Workshop on Research Issues in Data Engineering: Engineering E-Commerce/E-Business Systems (RIDE'02), page 7, Washington, DC, USA, 2002. IEEE Computer Society.
5. C. Peltz. [2003] Web services orchestration – A review of emerging technologies, tools, and standards. [http://devresource.hp.com/drc/technical white papers/WSOrch/WSOrchestration.pdf](http://devresource.hp.com/drc/technical%20white%20papers/WSOrch/WSOrchestration.pdf).
6. Joseph Chiusano. [2003] UDDI and ebXML Registry: A Co-Existence Paradigm. <http://www.oasis-open.org/archives/regrep/200304/pdf00000.pdf>, March 2003.
7. Alonso, G., et. al., (2003). Web Services – Concepts, Architectures and Applications. Springer Verlag, Heidelberg, Germany.
8. UDDI.org, [2002] “UDDI Version 2.03 Data Structure Reference”, Retrieved from <http://uddi.org/pubs/DataStructure-V2.03-Published-20020719.htm>, 2002.
9. UDDI.org, [2006] “UDDI Technical White Paper”, Retrieved April 30, 2006 from <http://uddi.org/pubs/uddi-tech-wp.pdf>
10. UDDI.org, [2006] “UDDI Version 3.0.2”, Retrieved April 30, 2006 from <http://uddi.org/pubs/uddi-v3.0.2-20041019.htm>
11. UN/CEFACT, OASIS. ebXML Website. <http://www.ebxml.org/>.