ANNALS OF THE FACULTY OF ENGINEERING HUNEDOARA

2004, Tome II, Fascicole 2

# BUSINESS PROCESS AND EBXML - WEB SERVICES INTEGRATION PLATFORM, REQUIREMENTS, ARCHITECTURES, SECURITY

[1] Carmen RĂDUŢ, [2]Maria STĂNILOIU

[1] Universitatea "Constantin Brâncoveanu" – PITEŞTI
[2] Universitatea "Constantin Brâncoveanu" – BRAILA

**ABSTRACT**

ebXML is a set of specifications that together enable a modular, yet complete electronic business framework. This session looks at various specifications that are being developed as part of the ebXML initiative and considers their relevance to designing an automated B2B collaboration. As part of designing the B2B collaboration, all the pieces in the puzzle are put together by using the modules that are part of ebXML.

Architecting a complex automated B2B collaboration involving multiple trading partners can be a very daunting job. This session presents the various steps involved in designing and implementing XML message-based collaboration layered on the ebXML framework. As part of implementing this architecture, various APIs for the JavaTM platform, available now and under development, will be utilized to access functionality presented by different layers.

This session offers you a look at a real-life business scenario with multiple layers, including the business process integration, partner profile management, registry/repository, and XML messaging. This session provides you with a detailed introduction to ebXML and the APIs for the Java platform available to implement this framework.

**KEYWORDS:**

*Business process and ebXML - architectures, web services*

## 1. INTRODUCTION

The larger organizations have been engaged in defining and implementing their systems around EDI (Electronic Data Interchange). EDI has allowed for not only the capture of common data-interchange formats but it has also tackled the challenge of capturing those formats, i.e. messages, by defining the Business Processes in which they are used. EDI though, has proved to be expensive not only due to the high network infrastructure setup/running costs but also the high costs of System Integration. For the existing EDI Implementations out there, there is usually a dominant business entity that has tried to enforce proprietary integration approaches on all the other partners. As a result of this, several companies (especially smaller organizations) have gone about building their electronic businesses and collaboration with their trading partners in an ad-hoc manner. Listed below are some of the points:

- Define Common Business Transactions e.g. Sending a Purchase Order
- Define Common Data-Interchange Formats i.e. Messages in the context of the above Transactions
- Define a mechanism for listing your organization's capabilities and the business transactions that your organization can perform in a common Repository accessible to all other organizations. In short, an ability to describe your Company Profile.
- Define a mechanism to allow organizations to discover companies and lookup their profile.
- Define a mechanism that allows two organizations to negotiate on the business terms before they commence transactions.
- Define a common transport mechanism for exchanging messages between organizations
- Define the security and reliability framework

It is important to have a common standard in order to make the global electronic commerce a reality, and this is the vision of ebXML (electronic business XML).

### *1.1. About ebXML*

ebXML is a global electronic business standard that is sponsored by UN/CEFACT (United Nations Center For Trade Facilitation And Electronic Business) and OASIS (Organization for the Advancement of Structural Information Standards).

The vision of ebXML[1] is to enable a global electronic marketplace where enterprises of any size and in any geographical location can meet and conduct business with each other through the exchange of XML-based messages.

[1] **ebXML terminology:**

- **Registry:** A central server that stores a variety of data necessary to make ebXML work. Amongst the information a Registry makes available in XML form are: Business Process & Information Meta Models, Core Library, Collaboration Protocol Profiles, and Business Library. Basically, when a business wants to start an ebXML relationship with another business, it queries a Registry in order to locate a suitable partner and to find information about requirements for dealing with that partner.
- **Business Processes:** Activities that a business can engage in (and for which it would generally want one or more partners). A Business Process is formally described by the Business Process Specification Schema (a W3C XML Schema and also a DTD), but may also be modeled in UML.
- **Collaboration Protocol Profile (CPP):** A profile filed with a Registry by a business wishing to engage in ebXML transactions. The CPP will specify some Business Processes of the business, as well as some Business Service Interfaces it supports.
- **Business Service Interface:** The ways that a business is able to carry out the transactions necessary in its Business Processes. The Business Service Interface also includes the kinds of Business Messages the business supports and the protocols over which these messages might travel.
- **Business Messages:** The actual information communicated as part of a business transaction. A message will contain multiple layers. At the outside layer, an actual communication protocol must be used (such as HTTP or SMTP). SOAP is an ebXML recommendation as an envelope for a message "payload." Other layers may deal with encryption or authentication.
- **Core Library:** A set of standard "parts" that may be used in larger ebXML elements. For example, Core Processes may be referenced by Business Processes. The Core Library is contributed by the ebXML initiative itself, while larger elements may be contributed by specific industries or businesses.
- **Collaboration Protocol Agreement (CPA):** In essence, a contract between two or more businesses that can be derived automatically from the CPPs of the respective companies. If a CPP says "I can do X," a CPA says "We will do X together."
- **Simple Object Access Protocol (SOAP):** A W3C protocol for exchange of information in a distributed environment endorsed by the ebXML initiative. Of interest for ebXML is SOAP's function as an envelope that defines a framework for describing what is in a message and how to process it.

In my opinion, ebXML will succeed in becoming universal by incorporating into the specifications more and more of what businesses do anyway as much as it will by actually getting businesses to do business differently. An illustration **(fig. 1, 2)** based on the ebXML Technical Architecture Specification will probably go a long way toward sorting out what ebXML means for business.

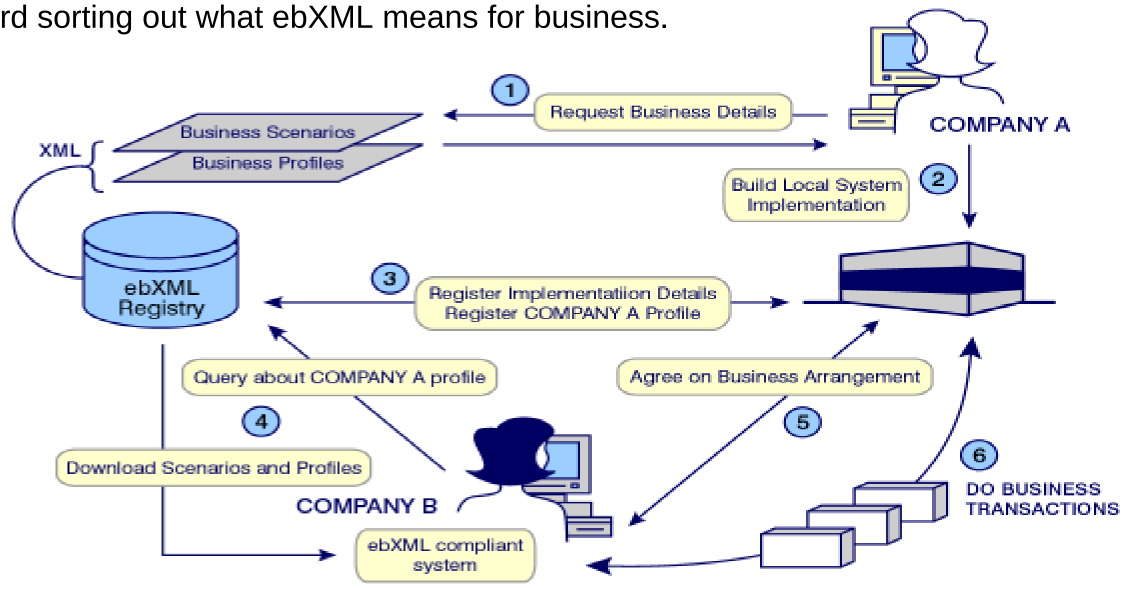
1
Request Business Details
COMPANY A
XML
Business Scenarios
Business Profiles
Build Local System Implementation
2
ebXML Registry
3
Register Implementatiion Details
Register COMPANY A Profile
Query about COMPANY A profile
Agree on Business Arrangement
4
5
6
Download Scenarios and Profiles
COMPANY B
DO BUSINESS TRANSACTIONS
ebXML compliant system

*Fig. 1. High-level overview of ebXML interaction between two companies*

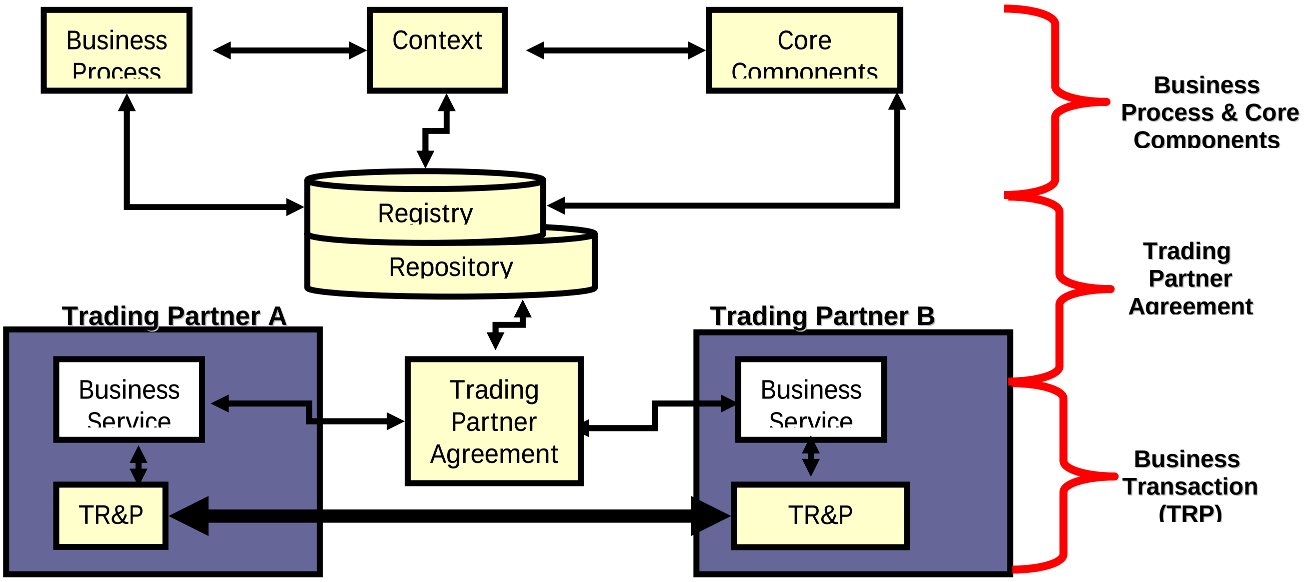

***Fig. 2.*** *ebXML Overview*

ebXML thus defines a framework for global electronic business that will allow businesses to find each other and conduct business based on well-defined XML messages within the context of standard business processes which are governed by standard or mutually-negotiated partner agreement. The ebXML standard addresses each of the above points, as we shall see in the next section. We shall now take a look at how a business would get itself ready to perform business transactions with other organizations, based on the ebXML Standard. Shown below are 3 key phases in the order in which they are supposed to be executed towards meeting that goal:

Implementation Phase **(fig. 3.)**, Discovery of Partner Information and Negotiation Phase **(fig. 4.)**, Transaction Phase **(fig. 5.).**

In the next diagram **(fig. 3.)**, the first thing to note is the ebXML Repository. This repository contains industry defined Business Processes and Scenarios that are commonly applicable to most business transactions. Companies can choose to extend these processes and add scenarios of their own. The repository also contains profiles for businesses that have already registered themselves for performing ebXML transactions with other trading partners. For an example Organization A, which is interested in doing electronic business as per the ebXML standard, consists of 3 steps as shown in the diagram above: Request Information, Implement ebXML System, Publish Business Profile

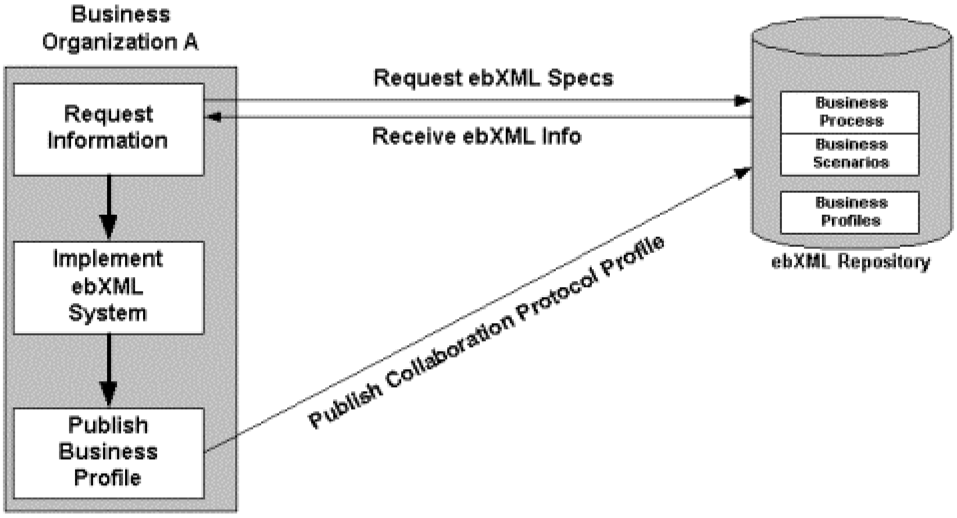

***Fig. 3.*** *Implementation Phase*

The first step towards that is to request the ebXML Specs (Business Processes, Business Scenarios) and understand them. Once the organization has taken a look at the specs, it decides which business processes it would like to implement, following which it needs to implement a system in-house based on those standards. It could either build a new system or build on top of an existing legacy system. The whole idea is to expose a system that understands and talks ebXML. There are several choices available today in the form of third-party applications that can just take and assemble together an ebXML System. Once the system is built, the organization is ready to conduct business with other organizations. To facilitate that, it needs to publish its profile known as a Collaboration Protocol Profile (CPP) to the ebXML Repository for other organizations to discover. A CPP thus enables any organization to describe its profile i.e. which business processes it supports, its roles in that process, the messages exchanged, the transport mechanism for the messages. Once the CPP is published to the ebXML Repository, it will allow other organizations to access it and learn about the capabilities of Organization A. At any time, Organization A is free to access its own profile, review and make changes as necessary.

We see in the next phase **(fig. 4.),** how Organization A readied itself for electronic business on the ebXML standard by first implementing the ebXML System in-house and then publishing its profile, which described its capabilities to the ebXML Repository. In this phase, we shall look at how Organization A does electronic business with a partner Organization B. As our Organization A has published its profile, Organization B has done the same. So the first step that Organization A does is to retrieve Organization B's profile information from the ebXML Repository. Once it has the profile, it is in a better position to understand Organization B's capabilities i.e. whether it supports the business processes that it is interested in, the messages to be exchanged, transport mechanisms, security and reliability of the process, etc. In the real world, businesses always negotiate terms and implement business contracts before conducting any business. ebXML is no different in that regard. So, the next step for Organization A is to send over a business contract called a Collaborative Partner Agreement (CPA), in ebXML, to Organization B. The CPA will be a reflection of the profile (CPP) of both the organizations. Both the organizations can now collaborate on the CPA and refine it to meet the business needs of both the organizations. Finally, both parties accept the agreement.

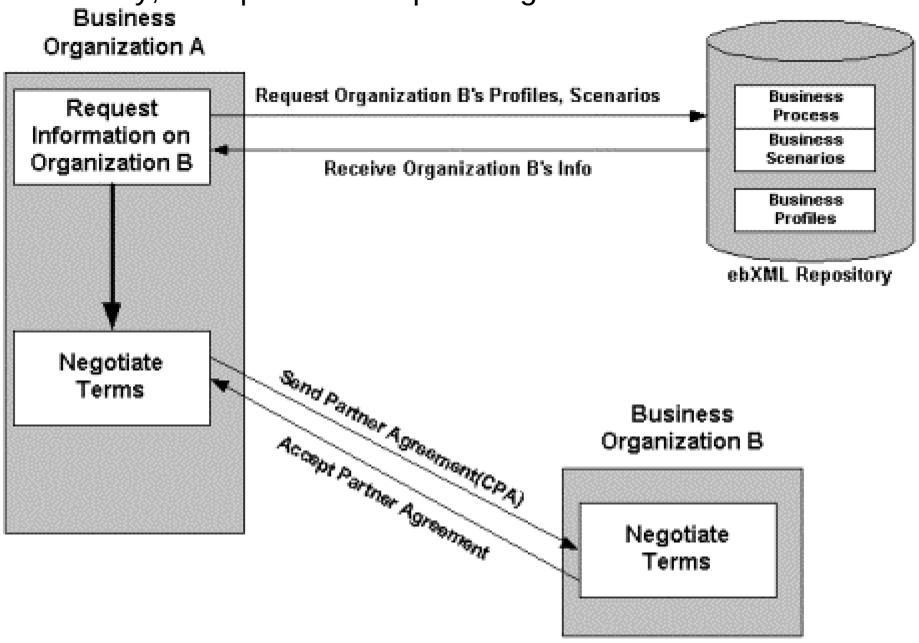

***Fig. 4.*** *Discovery of Partner Information and Negotiation Phase*

During this phase, it is very likely that key personnel from both organizations will meet in person and make assessments before committing to an eBusiness relationship.

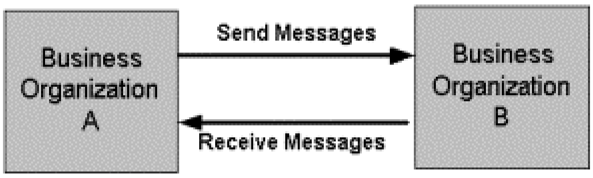

***Fig. 5.*** *Transaction Phase*

We are now ready to conduct transactions **(fig. 5.)** . A CPA was accepted in the previous phase and the transactions can be conducted in a pre-defined fashion where each business organization plays a pre-determined role in the transaction. The transactions consist of ebXML messages, which are sent over the standard ebXML Messaging Service.

### *1. 2. ebXML as Web Services Framework*

As e-business continues to develop, various technologies associated with computing underlie its evolution. Currently, the Java™ programming language and platform, the Extensible Markup Language, and transcoding are emerging as major technologies for performing e-business functions. In this overview essay, trends in these technologies are described, indicating how they will lead to future Web services.

In this overview, I would like to describe four major trends that I see coming along for the emerging e-business technologies of the Java** programming language, Extensible Markup Language (XML), and transcoding:

1. Continued integration of Java and XML into robust middleware such as the IBM WebSphere* software platform
2. Continued and accelerated standardization of Java and XML technologies for infrastructure and industries
3. Use of transcoding and XML technologies to support a much wider range of clients of every description, both synchronous and asynchronous
4. The move from tightly coupled applications to loosely coupled Web services

The Evolution of Integration is present in **fig. 6**:

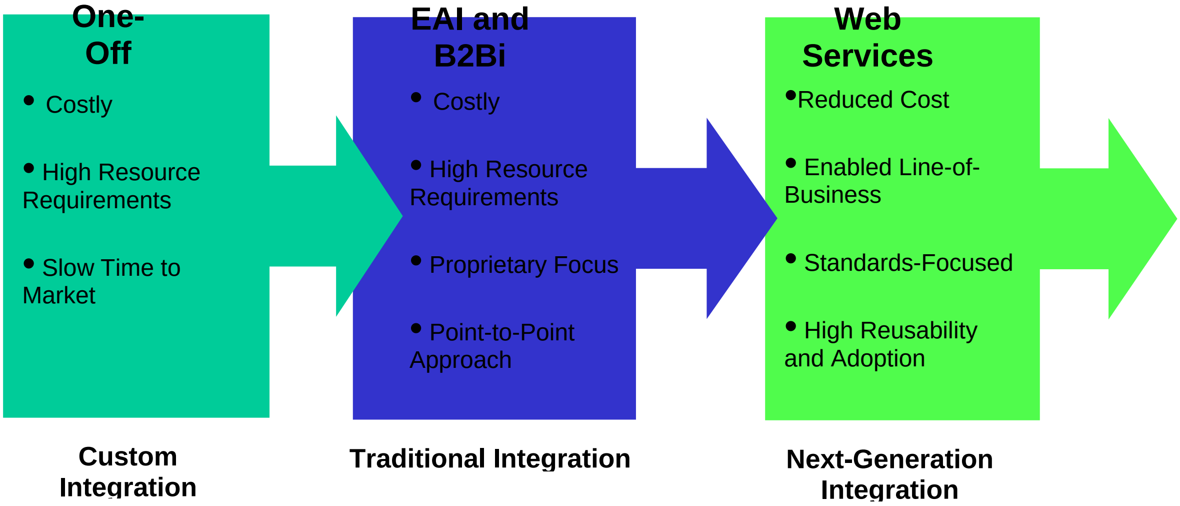
One-Off
• Costly
• High Resource Requirements
• Slow Time to Market
EAI and B2Bi
• Costly
• High Resource Requirements
• Proprietary Focus
• Point-to-Point Approach
Web Services
•Reduced Cost
• Enabled Line-of-Business
• Standards-Focused
• High Reusability and Adoption
Custom Integration
Traditional Integration
Next-Generation Integration

***Fig. 6.** The Evolution of Integration*

•Web services are the future of the Web because:

–Direct access to applications, programs and databases
–Browsers not required

•Web services are loosely defined as:

–Based on SOAP, WSDL, and UDDI
–An Internet address that maps XML documents to programs
–Web access to software-based services
–Anything that exposes programs to the Web!

**Types of Web Services are:**

•Simple: RPC based –Supports synchronous exchanges
•Compound: Aggregated “Simple” WS
•Complex: Conversational/Message-based

–Supports loosely coupled asynchronous exchanges
–Required for Enterprise Web Services
–Requires Enterprise QoS

**Simple Web Services:**

•Typically stateless
•Appears as an URI to the client application.

- The interaction centers around a service-specific interface
- Tightly coupled and synchronous, meaning response without request context is meaningless

**Complex Web Services:**

- Loosely coupled and document-driven
- Client invokes a message-based Web Service by sending it an entire document, such as a purchase order, rather than a discrete set of parameters
- The Web Service accepts the entire document, processes it, and may or may not return a result message
- Promotes a looser coupling between requestor and provider

A web service is a software application or component identified by a URI, whose interfaces and binding are capable of being described by standard XML vocabularies and that supports direct interactions with other software applications or components through the exchange of information that is expressed in terms of an XML Infoset via Internet-based protocols .

### *1.3. Security in ebXML Messaging*

*Elements of Security are:*

- **P**rivacy –Protect against information being disclosed or revealed to any entity not authorized to have that information
- **A**uthentication –Authenticate the claimed identity of the originator of a data item
- **A**uthorization –Protect against the threat that unknown entities enter into a system and ensures that an entity performs only authorized actions within the system
- **I**ntegrity –Protect against the threat that the value of a data item might be changed in a way that is inconsistent with the recognized security policy
- **N**on-repudiation –Protect against one party to a transaction or communication later falsely denying that the transaction or communication occurred

Security can be applied to: Transports (SSL, IPSEC), Messages (S/MIME, PGP), Systems

## 2. CONCLUSION

In this article we have seen what ebXML is all about. The article also highlights the different phases involved in getting an organization to conduct electronic business based on the ebXML Standard. We have all heard of and how businesses are trying to understand Web Services to achieve more dynamic and inter-operable applications between themselves and their trading partners. Several such organizations are now at an intriguing phase of their life where they are trying to understand how best to be a player within a global standard like ebXML.

The approval of ebXML specifications is moving along at a fairly rapid pace (certainly for a standards organization). The draft specifications were approved as version 2.5 recommendation in June, 2003. I suspect that it will take another year or two to shake out all of the issues and details for such an ambitious vision. It appears, however, that ebXML is on the way to widespread use a few years down the road. Now is the time, therefore, for businesses to begin a serious consideration of their own ebXML implementation plans.

“ebXML is our only chance this decade to establish an international e-commerce standard.”

## 3. REFERENCES

[1.] Clark, Jim; u.a.: ebXML Business Process Specification Schema Version 1.01. UN/CEFACT und OASIS, 2001, p. 230

[2.] Eric Chiu, "ebXML Simplified – A Guide to the New Standard for Global E-Commerce", 2002, p. 265

[3.] Alan Kotok & David R.R. Webber, "ebXML", September 2001, p. 330

[4.] Commerce Extensible Markup Language (cXML) Users Guide, Version 1.2, Ariba Inc., 2001.

[5.] XML Common Business Library (xCBL) Element and Datatype Structure Reference, Version 3.0, Commerce One, Inc., 2001.

[6.] Open Catalog Interface (OCI), Release 2.0 B, SAP AG, 2000.

[7.] Extensible Markup Language (XML), 1.0, World Wide Web Consortium, 1998.

[8.] Collaboration-Protocol Profile and Agreement Specification, Version 1.0, UN/CEFACT and OASIS, 2001.

[9.] ebXML Business Process Specification Schema, Version 1.01, UN/CEFACT and OASIS, 2001.

[10.] Web Services Description Language (WSDL) 1.1 Specification, World Wide Web Consortium, http://www.w3.org/TR/wsdl (2001).

[11.] Common Object Request Broker Architecture (CORBA), Object Management Group (OMG), http://www.omg.org/.

[12.] Universal Description, Discovery, and Integration (UDDI) Programmer's API specification,Version2.0,UDDI.org,http://www.uddi.org/pubs/ProgrammersAPI-V2.00- Open-20010608.pdf (2001).

[13.] Universal Description, Discovery, and Integration (UDDI) Technical White Paper, UDDI.org, http://www.uddi.org (2000).

[14.] Open Applications Group (OAG), http://www.openapplications.org/.

[15.] OASIS/ebXML Registry Services Specification v2.5. Committee Approved Specification. Produced by the OASIS/ebXML Registry Technical Committee. June 2003. 189 pages.

[16.] OASIS/ebXML Registry Information Model v2.5. Committee Approved Specification. Produced by the OASIS/ebXML Registry Technical Committee. June 2003. 94 pages.

[17.] OASIS/ebXML Registry Services Specification v2.5 [source]

[18.] OASIS/ebXML Registry Information Model v2.5 [source]

[19.] OASIS/ebXML Registry Technical Committee website.

[20.] "... the OASIS ebXML Registry Services Specification 2.4 and the OASIS ebXML Registry Information Model Specification 2.4 have been approved by the TC as TC approved specs.

[21.] "OASIS ebXML Registry Technical Committee Releases Approved Version 2.0 RIM/RS Specifications." News story 2001-12-12.

[22.] "Electronic Business XML Initiative (ebXML)" - Main reference page.

[23.] http://www-3.ibm.com/software/webservers/commerce/wc_be/.

[24.] http://www-3.ibm.com/software/webservers/commerce/wcs_me/.

[25.] www.ebxml.org/specs/ebMS.pdf

[26.] www.w3c.org/TR/SOAP-attachments

[27.] www.w3c.org/2000/xp/Group

[28.] www.oasis-open.org/committees/ebxml-msg

[29.] www.ws-i.org (NOT www.wsi.org)