

# A Tool for the Management of ebXML Resources

S.S. Sooriarachchi, G.N. Wikramanayake, G.K.A. Dias  
University of Colombo School of Computing

E-mail: sumekhala.sooriarachchi@ifs.lk

## Abstract

*The ebXML initiative is designed for electronic interoperability, allowing businesses to find each other, agree to become trading partners and conduct business. ebXML repository is used to store the ebXML resources and the ebXML registry is used to discover these resources. ebXML resources may be in the form of XML documents, Document Type Definitions, XML Schemas, UML models and various other forms. Therefore a proper tool is required for the management of these resources.*

*Management of ebXML resources has to be done by the experts who create and modify and are in charge of these resources. If there is a tool for the management of these resources, the experts can concentrate more on the content of these resources rather than management of them.*

*This paper describes the implementation of a graphical tool for the management of ebXML resources based on the proposed ebXML specifications. Two approaches are being used to discover these resources. Registry Navigator is one, which is in a tree structure and gives a full view of the registry. Query Manager is the other, which enables quick referencing to the resources if the user is fully aware of what resources are needed. The documents discovered in this manner are to be opened in the Editor pane of the tool. This editor facilitates the creation and modification of documents easily.*

*The tool also incorporates capabilities to add and remove resources through the Life Cycle Manager. It also has a mechanism, which keeps track of different versions of the resources, so that the preferred version can be referred with minimum effort. Finally, it allows non-expert users to dynamically access the content of the registry over the web.*

**Keywords:** ebXML Resources, Registry, Repository, Graphical Tool

## 1. Introduction

Today the world of electronic collaboration [5] is developing rapidly, introducing new technologies, and new ways of collaborating. The success of collaboration will depend on the ability of a corporation to make sure that their applications are not only dynamic, but maintain a high degree of inter-operability with collaboration partners.

Electronic Data Interchange (EDI) [4] essentially defined the technology of electronic collaboration for the last millennium, but its popularity is waning for a variety of reasons. EDI is an expensive solution, due to the high cost of network infrastructure and system integration. It has also proven to be complex, difficult to maintain and inflexible in the face of changing market conditions. Smaller businesses with low volume collaboration needs, simply preferred to stay away from EDI.

The challenge to achieving dynamic e-business collaborations lies in the need to have a low cost, flexible software solution that allows corporations to build new applications in response to changing business needs while adhering to a defined electronic business standard.

Web services [9, 11] offer the potential for seamless application integration regardless of programming language or operating environment. Web services technology is based on a set of existing Internet standards and widely accepted specifications: HTTP, XML, SOAP, WSDL and UDDI. Web services alone are insufficient to achieve effective electronic collaboration unless Web services are applied in the context of collaboration standards such as ebXML [3].

ebXML is a set of specifications that enable a modular, yet complete electronic business framework. If the Internet is the information highway for electronic business, then ebXML can be thought of as providing the rules of the road. The ebXML initiative is designed for electronic interoperability, allowing businesses to find each other, agree to become trading partners and conduct business [7]. ebXML brings EDI's benefit of a common framework for conducting business in a supply chain

management model [15] to the small-to-medium-sized enterprise and to those enterprises that require a more flexible, loosely coupled e-commerce infrastructure.

### 1.1. Management of ebXML Resources

Even if the ebXML specifications exist, the benefits would not be fully realized if they are not properly managed or if they cannot be discovered as and when they are needed. Registry and repository is the mechanism to register and discover company and business service profiles, as well as business process specifications with related message exchanges, and other XML and e-commerce resources.

Early adoption of XML by industry partners is creating opportunity for information reuse and collaborations over the Web. At the same time, the rapid emergence of XML Document Type Definitions (DTDs) and vocabularies from industry and government sectors has focused attention upon issues of resource identification, classification, cataloging and delivery that hinder reuse and interoperability. The results of new collaborative endeavors are not necessarily easy to identify and access on the Internet.

For ebXML resources, registry/repository acts as a central warehouse. It is used to submit, store, retrieve and manage resources to facilitate ebXML-based business-to-business (B2B) partnerships and transactions. Submitted information may be, for example, in the form of business profile information, XML schema and documents, business process definitions.

Also there should be a discovery mechanism for businesses to find and engage one another. Registering a business service profile and business process schema in an ebXML registry/repository enables them to be located.

Since every company engaged in e-business concentrate on the growth of their individual enterprises, a central body is needed to initiate and be responsible for management of ebXML resources. Especially in Sri Lanka, where the businesses are still at the dawn of e-collaborations, it is better to have a responsible organization that can initialize and direct trading partners to use ebXML resources. This responsible central organization can build a comprehensive registry covering many areas of business giving easy access to these resources, which will facilitate the specifications to get a better recognition and a quick adoption. The work presented here is part of achieving such a goal.

## 2. Registry/Repository

### 2.1. Concept of Registry and Repository

Registries are aimed to manage the challenge of passing consistent information between a business system and its' suppliers and customers. Associated with this is the critical need to provide the means to accurately and quickly locate specific information on a topic or a domain.

Discovering new trading partners and the rules for engaging in a particular line of business are clearly powerful reasons for accessing e-business registries. But there are other functions designed to meet the needs of cost effective and timely use of information.

Essentially the functionality of an e-business registry can be divided into three broad domains.

- Providing a directory of members and services available with search and discovery.
- Providing human readable technical documentation and specifications organized using applicable domain classifications and categories.
- Enabling automated machine to machine e-business interactions through machine readable consistent content and process definitions, associations and linkages.

The information that can be discovered via the registries actually resides in the repositories. A repository is not just a passive data dictionary or database. It is an integrated holding area that should also keep the information up to date by providing processing methods and make it available to a user as needed. A repository, which maintains valuable information about all of the information system assets of an organization and the relationships between them, acts as a central manager of all of the information resources in an enterprise. A repository should provide services such as change notification, modification tracking, version management, configuration management, and user authorization [10].

The widespread availability of XML-capable clients and their flexibility in structuring information make it possible for XML to become the universal data format. Without the help of a repository, it will be difficult to control XML objects in a manageable way and make them available when needed.

XML repository provide several basic functions such as importing/exporting XML data from original text files, user check in/check out, version control, as well as searching and querying on repository items (XML documents). In the electronic commerce world, XML repositories are the online source for

obtaining the appropriate tag, document-type definition, data element, database schema, software code or routines. As a result, companies, especially small enterprises, can speed up processing and expand their ability to conduct electronic commerce [10].

## 2.2. ebXML Registry/Repository

ebXML registry/repository acts as a central warehouse for ebXML resources. It is used to submit, store, retrieve and manage resources to facilitate ebXML-based business-to-business partnerships and transactions. Submitted information may be, for example, in the form of business profile information, XML schema and documents, business process specifications, business context descriptions, Unified Modeling Language (UML) models, business collaboration information or even software components. Runtime artifacts of ebXML are shown in Figure 1.

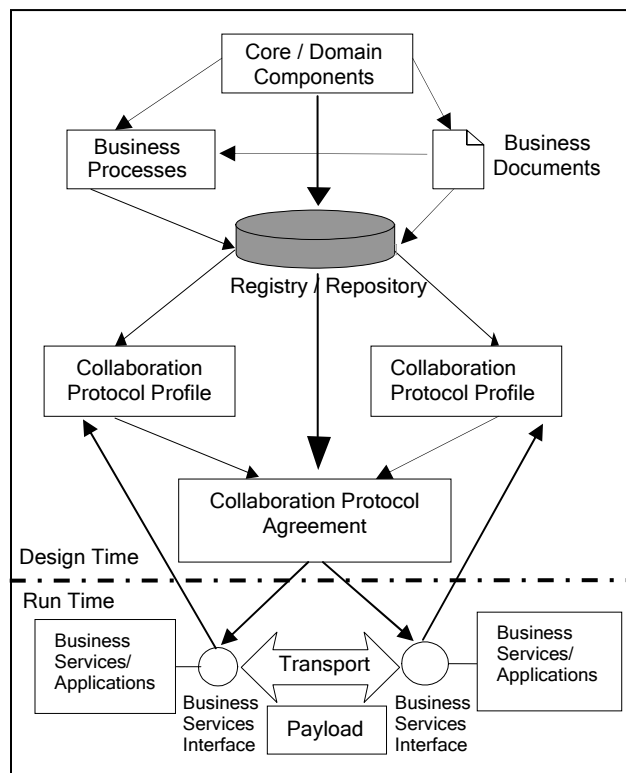


Figure 1: ebXML runtime artifacts [5]

A variety of XML formats are used in ebXML, such as:

- Business Process Specifications (BPS)
- Business Document Specifications (BDS)
- Collaboration Protocol Profiles (CPP)
- Collaboration Protocol Agreements (CPA)
- Log/Audit Trail interchange

- Core Components definitions
- UMM Models

Business processes represent the *verbs* of e-business. To enable integration of business processes within or between businesses, clear definitions of the business processes must be expressed in such a way, that they are understandable by the people and software of other businesses or business units.

ebXML defines a methodology for modeling business processes as a set of choreographed document exchanges, as well as the procedure for representing business process transactions in unambiguous ebXML business process schemas. These collaboration models make no demands on the underlying infrastructure. Consequently, businesses and industry organizations can develop and reuse business processes, without concern for the specific platform or software application that will execute the transactions.

Once a business process is defined, businesses need a standard means of describing the roles in which they are prepared to engage for that business process, as well as the technical capabilities they support to fulfill those roles. Generally, the description is defined in terms of roles such as *buyer* and *seller*. The CPP identifies which role or roles the party is capable of playing in each collaboration protocol referenced by the CPP.

CPP describes a partner's IT capabilities. These capabilities include what communication protocols (HTTP, SMTP, FTP etc.) they support, what security requirements they place upon the message exchanges, and what business processes they support. A CPP describes all the things a partner can do.

The protocol profile contains information about the business collaborations that a company supports and its message exchange capabilities. Using information from these profile documents, a collaboration agreement is formed to define the way in which parties will interact in the performance of business collaborations. A clearly defined trading agreement must be created that can be used to govern the transactions between partners.

A CPA first identifies the parties to the agreement. CPA contains following details:

- Communication protocols the parties will support.
- The messaging protocol to be used in exchanging business documents.
- Information needed to ensure a secure interchange of information between the parties.
- Business Transactions or services that the parties agree to interchange.

## 2.3. ebXML Specifications

The participants of ebXML had a vast amount of experience in various industries, EDI and XML standards and initiatives. They were able to bring their wealth of knowledge and experience to develop a set of specifications.

The ebXML framework consists of the following specifications [6]:

- ebXML Technical Architecture Specification
- Business Process Specification Schema (BPSS)
- Registry Information Model (RIM)
- Registry Services Specification (RSS)
- ebXML Requirements Specification
- CPP and CPA Specification (CPPA)
- Message Service Specification

"The RIM [12] provides a blueprint or high-level schema for the ebXML Registry. Its primary value is for implementers of ebXML Registries. It provides these implementers with information on the type of metadata that is stored in the Registry as well as the relationships among metadata Classes."

The RIM defines:

- Types of objects that are stored in the Registry
- How stored objects are organized in the Registry

A set of Registry Services that provide access to Registry content to clients is defined in the ebXML RSS [13]. The RSS defines the interface used to the ebXML registry as well as interaction protocols, message definitions and XML schema. The registry services permits access to the repository or content management system.

RSS assumes B2B exchanges that are carried out in the following sequence:

- BPS are submitted
- Business Process Documents are submitted
- Seller's CPP is submitted
- Buyer discovers the seller
- CPA is established after negotiations
- Once the seller accepts the CPA, the parties may begin to conduct B2B transactions

This specification also defines the actors who may interact with the registry, such as Registry administrator/Responsible organization, Registry user, Registry guest, Submitting organization (same as Registry administrator).

The ebXML Registry Service is comprised of a robust set of interfaces designed to fundamentally manage the objects and inquiries associated with the ebXML Registry. The two primary interfaces for the Registry Service consist of:

- A Life Cycle Management interface that provides a collection of methods for managing objects within the Registry.
- A Query Management Interface that controls the discovery and retrieval of information from the Registry

A registry client program utilizes the services of the registry by invoking methods on one of the above interfaces defined by the Registry Service.

## 2.4. Registry/Repository Systems

### 2.4.1. Sun ebXML Registry and Repository

The Sun ebXML Registry/Repository Implementation (RegRep) [17] can be used to submit, store, retrieve, and manage resources to facilitate ebXML-based B2B partnerships and transactions.

The RegRep implementation is based on open, non-proprietary, platform-neutral J2EE technology. What this means is that you can use the development tools, application servers, databases, and platforms you want. Core components of this implementation include a Registry Information Model, Registry Services, Security Model, Data Access API, Java Objects Binding Classes and JSP Tag Library.

### 2.4.2. OASIS XML Repository

The XML interoperability consortium OASIS has announced public access to the first phase of XML.org Registry, an open registry and repository for XML specifications and vocabularies [18].

The site is designed to both a central registry for XML schemas and other public resources (DTDs, namespaces, stylesheets, public key certificates), and an open development forum for designing useful repository/registry architectures.

The XML.ORG Registry was developed by Documentum and Sun Microsystems using software components from Documentum, iPlanet, and Oracle. Documentum 4i eBusiness edition, the content management platform powering the registry application, drives the entire process from the submission of a schema to its availability for public access via Documentum Site Delivery Services.

### 2.4.3. IBM XML Registry/Repository

The IBM XML Registry/Repository (XRR) [8] is "a data management system that manages and provides services for XML artifacts including schemes (DTD, XSD), stylesheets (XSL) and instance documents (WSDL). User can use XRR to obtain an XML artifact automatically, search or browse for an XML

artifact, deposit an XML artifact with or without related data, and register an XML artifact without deposit.

The registry provides a search of registered objects based on their metadata. Registry facilities include registration, search and retrieval of registered objects, and Administration.

The 'Repository' service "provides access to registered objects. Through the repository, a user can download a registered object using standard identifiers (URLs)." The current version of XRR runs on Windows NT, Windows 2000, Linux, AIX, and Solaris; it supports basic Servlet/JSP functionality. Databases: IBM DB2, Version 8, must be installed and running.

#### 2.4.4. CENTRAL Registry Project

CENTRAL registry project of Boeing enterprise [2] provide a company-wide resource for registering, locating, sharing, and re-using XML schemas, DTDs, and other information needed to enable the electronic interchange of data and for understanding the meaning of that data.

#### 2.4.5. RepoX XML Repository

RepoX, an XML repository [10], has been developed for the METEOR workflow system. It maps XML documents to a relational-object database and also provides extraction/retrieval, version control, check in/check out, and searching and query functions.

The RepoX repository provides full support for searching, querying, and versioning. An XML document can be modeled as a "rooted, directed, ordered, and labeled tree". To access and manipulate the XML document as a tree structure, the Document Object Model (DOM) core interfaces are used in the RepoX.

### 2.5. Graphical Tools and Web Interface

Since ebXML resources are stored in ebXML Registry Repositories there is a need for graphical user interfaces in order to be able to manage these resources. Our aim is to study the features required for an ebXML registry repository and then to design and implement a graphical tool and a web interface to manage these resources. For this purpose we have studied a number of graphical user interface techniques for general and specific management, editing, storage and validation of ebXML artifacts such as XML specification documents. Above listed registry repositories are part of this study.

### 3. Design of RRMS

The registry repository management system (RRMS) is designed in three main layers namely presentation layer, logic layer and the storage layer. The main architecture of the system is shown in the Figure 2.

The presentation layer contains the four interface components: *Life Cycle Manager* and the *Query Manager* with the navigator belonging to the standalone tool, and the web interface of the web based subsystem. Standalone tool is to be used only by the registry/repository admin of the responsible organization and the web-based system is for the trading parties from different industries.

The logic layer will take care of depositing the XML documents in its correct location in the repository, obtaining metadata from users about the documents to be saved and displayed in the registry. The storage layer physically represents the registry and repository. Design details of RRMS with class diagrams and sequence diagrams can be found in [16].

The repository is primarily categorized under different industries. Under each industry there would be a node for its' Business processes, CPPs of the trading parties of that industry and CPAs of the trading parties who get together for collaborations.

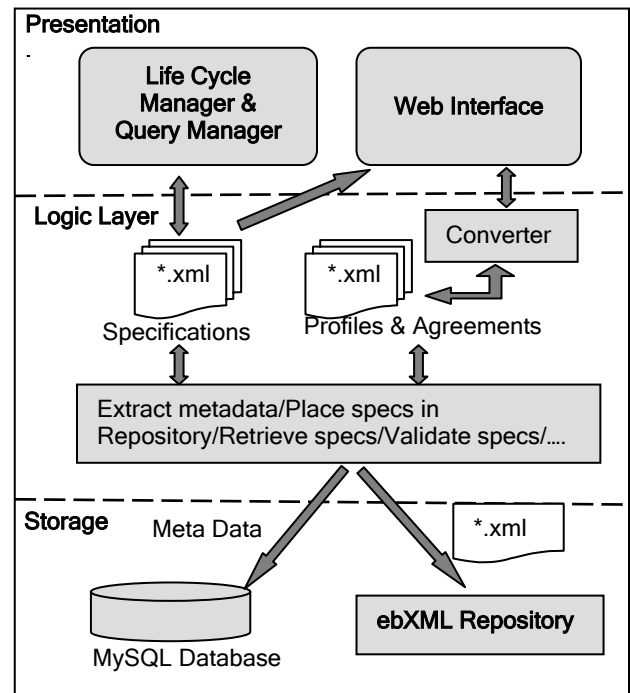


Figure 2: Main Architecture of RRMS

Generally there would be many business processes under one industry. Therefore the business

processes node has sub nodes for all the business processes corresponding to that industry.

A particular business process node further divides into BPS and BDS under this model. Since there would be more than one document that are exchanged between trading parties for a particular business process, there would be a node for each of the document specification.

In this repository structure, only the leaf nodes will carry repository items (i.e. specifications, profiles or agreements).

## 4. Implementation of RRMS

### 4.1. Environment

JBuilder 7 Enterprise Edition of Borland Software Corporation is used for the development of the graphical tool and the JSP pages of the web application. JBuilder contains major improvements in developer productivity, as well as a cleaner, more intuitive user interface and dramatic performance enhancements.

MySQL 4.0, the most popular Open Source SQL relational database management system, is developed, distributed and supported by MySQL AB. The MySQL Database Server is very fast, reliable, and easy to use. It also has a practical set of features developed in close cooperation with the users.

MySQL Connector/J 2.0.14 (Formerly MM.MySQL - Mark Matthews JDBC Driver for MySQL), which is a free product, is the JDBC driver used during the implementation.

### 4.2. Proof of Concept Implementation

This section gives a detailed description of the implementation phase of the registry/repository management system. As described under the design, only a selected number of resources are considered for the implementation. Implementation was carried out in the following steps.

#### 4.2.1. Creation of the database: Registry

As the first step, the database, which plays the role of the registry in this system, was created. This database contains data about the documents stored in the repository, and also a reference (URL) to the absolute location of these resources.

The database named *registry* is created in the MySQL database server containing the following tables. The MySQL command used in obtaining the database structure is also given here. Tables in the

*registry* are *agreements*, *bds*, *bpps*, *businessprocess*, *cpatemplates*, *cpptemplates*, *document*, *industry* and *profiles*.

Of the above, the tables *industry* and *businessprocess* and *bds* does not contain information about a specific document. They are needed for the relationships in the database and to maintain the repository structure.

#### 4.2.2. Creation of folder structure: Repository

It is in the folder structure, the resources are actually kept. When creating each of the documents, it also dynamically creates the path (URL) in which the document is to be kept, according to the options selected by the user. For e.g. if the creation of a new version of a business document spec is considered, code segment is as follows. The full URL is sent to the database (*document* table) to be used when retrieving the documents.

#### 4.2.3. Development of the Graphical Tool

Graphical tool plays an important part in the system. It is through this, the expert users manage the registry/repository. The main parts of the tool consist of registry navigator, life cycle manager, query manager, editor pane and web interface.

##### (a) Registry Navigator

The Registry Navigator was implemented using the Swing component, *Jtree*, which can be used to provide a view of hierarchical data. Like any non-trivial Swing component, the tree gets data by querying a data model. The tree in this case was placed in a scroll pane to allow easy navigation when the tree grows in size with the addition of more and more new resources to the registry/repository.

The tree is generated dynamically with the data from the appropriate tables in the database and by placing them in the tree so that it reflects the real folder structure of the repository. The tree is also refreshed, when a new resource is added, so it gives the most updated view of the registry all the time.

The documents, which are represented by the leaf nodes of the tree, can be opened in the Editor pane directly by selecting a document in the navigator and clicking on *Open* button (Figure 3).

##### (b) Life Cycle Manager

The Life Cycle Manager has to provide facility for the expert users to add resources to the registry/repository, modify these resources and finally remove these resources.

The addition of resources is done through the menu *Life Cycle Manager* of the graphical tool. How each of its menu items function is described below.

**New Industry:** Through this frame a new industry can be added to the database and the navigator when it is refreshed.

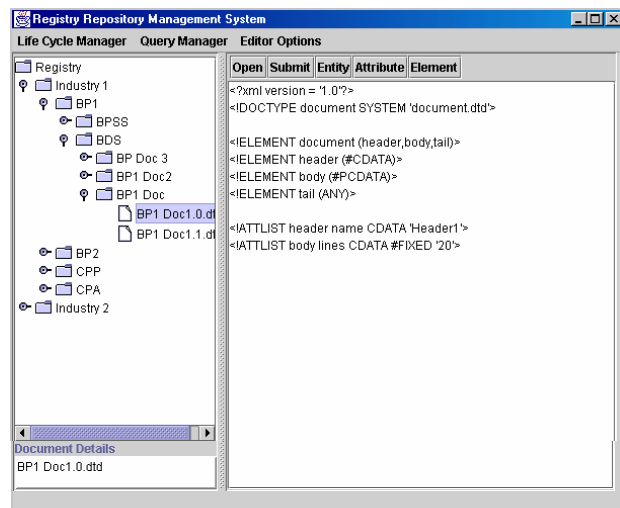


Figure 3: Registry navigator with a document opened

**New Business Process:** Through this a new business process is added to the database. The new business process is for a particular industry. Therefore, this frame enables selection of the required industry through a combo box. This combo box is filled with the industry names from the *industry* table.

**New BPSS:** Through this a new BPSS is added to the database. This is used to add a new business process specification schema for a selected business process of a selected industry. The available industries are added to the combo box by a query and the business processes are filled to another combo according to the selected industry.

**New BDS:** This has two sub menu items, one to add a New BDS type and the other to add a New Version. Here too, the new resource is added for a selected process of a selected industry. This selection is enabled through two combo boxes similarly to the above-mentioned methods. When a new BDS type is added *bds* table is updated and when a new version is added the 'document' table is updated.

**CPP and CPA:** Both the menu items function very much similarly. Only the tables that are queried and updated differ. For the CPPs *profiles* and

*cpptemplates* tables are used while *agreements* and *cpatemplates* are used for the CPAs. These are added to the registry, industry wise. When a new template is added it is sent to the *cpptemplates* table or to the *cpatemplates* table.

Removal of resources is enabled at a higher level, by giving the option to remove resources of a whole industry or an entire business process (Figure 4). When this is done, the related documents are also removed from the database according to the primary keys of the *industry* table and the *businessprocess* table (i.e. Delete is cascaded). Also removal of resources is enabled at an individual document level. That is, user can remove one document at a time. Before removing a resource, the system will display a dialog box to make sure that the removal is deliberate and not done by mistake.

In both these cases, the node has to be removed from the tree navigator and the corresponding records have to be removed from all the tables and finally the document has to be deleted from its exact location in the repository.

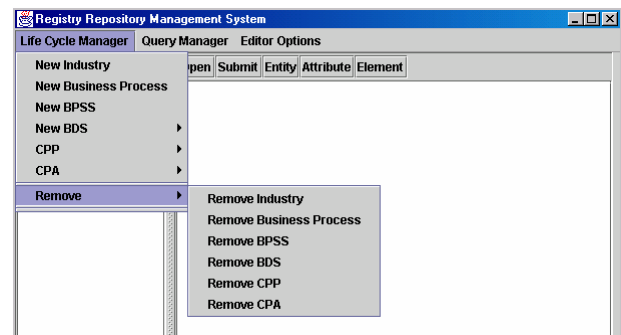


Figure 4: Removal of Resources

### (c) Query Manager

Query Manager allows the expert users to query for BPSS, BDS, CPP and CPA.

**Querying BPSS:** The Query BPSS has two combo boxes for the user to select the industry and the business process. The business process combo box is filled according to the industry selected. The BPSSs that match the selected options are retrieved from the table *bpss* and displayed in a table. If the user wants to view a BPSS document, selecting a row in the table and then clicking on *View BPSS* button enable this.

**Querying BDS:** The functionality of this option is very much similar to the above explained one. Only difference is that the table queried is *document*, rather

than *bpps*. Figure 5 is the interface to retrieve BDS documents.

**Querying CPP:** Query CPP tabulates the details about the CPPs submitted to the registry. The CPPs can be retrieved industry wise through this query manager by selecting the required row from the table. The CPP template also can be retrieved according to the selected industry. In here, the tables queried are *profiles* and *cpptemplates* to retrieve the documents stored in the repository. Functionality is similar to that of Querying BPSS.

**Querying CPA:** This is very much similar to the Querying CPP, except for difference in the tables queried. They are *agreements* and *cpatemplates* for the case of CPAs.

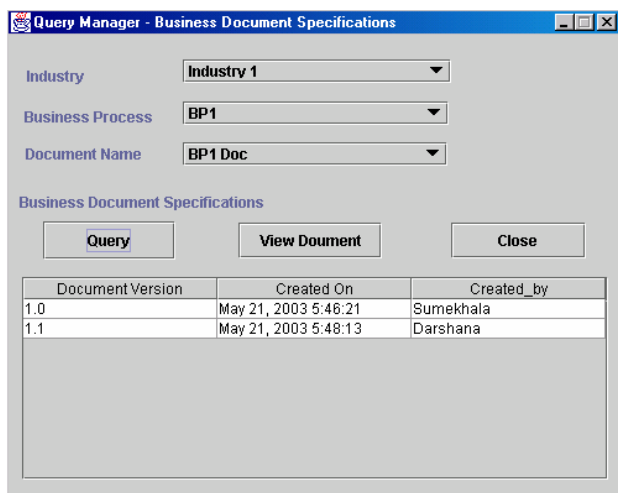


Figure 5: Accessing BDS using query manager

#### (d) Editor pane

Editor pane allows the expert users to add and modify the content of the resources created with the Life Cycle Manager menu options. First the documents have to be opened in the Editor pane. The URL of the currently opened file is kept in a String variable to be used when submitting the file back to the repository after modifying the content. To make it easy for the expert users to add content to the document files they are creating, the Editor Options menu gives some options to the users.

The options given are mainly to support creation of BDS documents. Since BDS documents are to be created in DTD format, the addition of building blocks of DTDs such as, a root, elements, attributes and entities are enabled through the sub-menu items.

#### (e) Web interface

The web interface is for the purpose of displaying the registry content and discovering BPS. Java Server Pages (JSP) was used to generate the dynamic web pages which provides easy navigation through the registry's resources.

Here also the *registry* database is queried according to industry and the business process selected by the user, by passing the primary key of *industry* table or *businessprocess* table as a parameter to the subsequent .jsp pages.

## 5. Evaluation

This section evaluates the achievements by comparing them with the predefined objectives for various stages of the project.

This project carried more work during the analysis and design stages. A comprehensive analysis had been carried out covering the issues related to e-business, e-business collaborations, standardizing e-business. Extra effort was put in understanding the ebXML framework and the role played by registry/repository. To understand the functions of registry/repository, a literature survey was carried out covering a number of related architectures. By putting together the information gathered as above, an in depth requirements analysis was undertaken achieving the objectives set.

During the design stage, architecture for the registry and repository had to be developed, along with the architecture for the tool as planned. According to the requirements and to reduce the complexity, the registry was designed as a database, which would store meta-data about the resources. The repository was designed as a folder structure as described in the design section.

The tool is designed to help the expert users, to manage the ebXML resources according to the defined scope of the project. The tool was designed to have a navigator for the purpose of discovering resources in the registry/repository. The resources are to be accessed and manipulated through the Life Cycle Manager and the Query Manager. These details are covered in the design section of this report. In this stage, in addition to the graphical tool required by the expert users, some other functionality, which are needed by the non-expert users such as submission of CPP and CPA are also designed having related future work in mind.

Proof of concept implementation for the above-mentioned design has four main parts. The Navigator in the form of a tree, which facilitates discovery of resources, the Life Cycle Manager to support the management of the resources through their life time,



the Query Manager for discovering and retrieving the resources and the Editor pane, which supports the creation, modification and viewing of the resources. Also a web application is implemented as expected, to discover the registry content, which are updated dynamically over the web.

The implementation of submission of CPPs and CPAs is not covered since it is out of scope of the targeted work. The design done in this area is expected to direct future work related to providing registry facilities to the non-expert users.

## 6. Conclusion

The Graphical Tool for the management of ebXML resources designed under this project was targeted towards the expert users who would be using the registry/repository, for the purpose of maintaining these resources.

The suitability of such a tool for the e-business community of Sri Lanka was a main concern. Since e-business is still at a primary stage in Sri Lanka, a centralized tool was thought to be more suitable, where initially one authority would maintain the ebXML resources through the resulting tool.

During the implementation of the tool, only a limited number of ebXML resources were taken into consideration, namely business process specification schemas, business document specifications, collaboration protocol profiles and agreements. And also limited Editor options were given to avoid 'reinventing the wheel'.

By developing an architecture for the registry/repository, and designing a graphical tool to manage the ebXML resources to be stored in the registry/repository with a supporting proof of concept implementation the objectives of the project have been achieved within the given constraints. This tool is expected to initiate standardization of e-business in Sri Lanka and to cater to the needs of its' fast growing e-community.

### 6.1. Future work

The Graphical tool for the management of ebXML resources developed under this project can be thought of as a starting point for a number of important areas related to the development of e-business in Sri Lanka.

The graphical tool developed under the project has further capacity for enhancements. Following are some suggestions for improvements of the tool and to make it more effective and comprehensive.

- The Editor pane of the tool, which is to be used by the expert users to create the documents, can be improved by providing a wide range of Editing options as in well-known XML editors

like 'XML Spy' [1]. Wizards for creating such documents would be a value adding option to the tool.

- The registry can be made more comprehensive to include resources of various types, other than the resources considered for the purpose of this project. These other resources can be UML models, Core Components, etc.
- For the registry to serve the business community of Sri Lanka, it should contain well-prepared standards (ebXML resources) for various industries. This has to be done after a well planned careful analysis of each industry and then creating and submitting these resources to the registry/repository through the tool. A dedicated team is proposed to do such analysis and for the management of the registry/repository, so that it would increase the confidence the business personnel has on the standards.

## Acknowledgements

This work was done at University of Colombo School of Computing are part of collaborative research with the OpenXML Laboratory, University of Stockholm. Valuable advice and guidance given by Mr. Anders W. Tell, Mr. Erik Perjons, Mr. Harsha Wijewardena and the members of the research group is acknowledged.

## References

- [1] ALTOVA - XML Development, Data Mapping, and Content Authoring, accessed on 071103, <http://www.xmlspy.com/>
- [2] Breininger Kathryn, "CENTRAL Registry Project", Boeing aviation integration, e-business solutions, Standard Services Group, 2001, accessed on 071103, <http://boeingicp.eep.gmu.edu/presentations/Kathryn%20Breininger%20-%20Central%20Registry%20Project.pdf>
- [3] ebXML - Enabling A Global Electronic Market, accessed on 071103, <http://www.ebxml.org>
- [4] EDI Standards, Federal Information Processing Standards Publication 161-2, 1996, accessed on 071103, <http://www.itl.nist.gov/fipspubs/fip161-2.htm>
- [5] Electronic Collaboration: A practical guide for educators, Brown University, 1999, accessed on 071103,

- <http://www.lab.brown.edu/public/pubs/collab/elec-collab.pdf>
- [6] Harvey Betty, "The Role of XML in E-Business", Electronic Commerce Connection Inc., 2002, accessed on 071103, <http://www2.cs.uregina.ca/~tang112x/research/papers/2003w/>
- [7] Ibbotson John, ebXML Trading-Partners Specification, Internationales Congress Centrum, XML Europe, Germany, 2001, accessed on 071103, <http://www.gca.org/papers/xml europe2001/papers/html/s09-2.html>
- [8] IBM alphaWorks Releases XML Registry/Repository Data Management System, 2001, accessed on 071103, <http://xml.coverpages.org/ni2001-06-04-a.html>
- [9] INCITS — the InterNational Committee for Information Technology Standards (formerly X3), accessed on 071103, <http://www.X3.org>
- [10] Minrong Song, John A. Miller and Ismailcem B. Arpinar, "RepoX: An XML Repository for Workflow Designs and Specifications", Technical Report #UGA-CS-LSDIS-TR-01-012, University of Georgia (August 2001) 43 pages, accessed on 071103, [http://chief.cs.uga.edu/~jam/home/theses/song\\_thesis/song\\_minrong\\_repoX.pdf](http://chief.cs.uga.edu/~jam/home/theses/song_thesis/song_minrong_repoX.pdf)
- [11] Morais Pravin, "Dynamic e-Business Using Web Service Workflow", SearchWebServices, New, June 2002, accessed on 150903, <http://www.cysive.com/news/062602.htm>
- [12] OASIS/ebXML Registry Information Model v2.1, Approved Committee Specification - OASIS/ebXML Registry TC, 2002.
- [13] OASIS/ebXML Registry Services Specification v2.1, Approved Committee Specification - OASIS/ebXML Registry TC, 2002, accessed on 071103, <http://www.oasis-open.org/>
- [14] Open ebXML Laboratory- project catalog, accessed on 071103, <http://www.openebxmlab.org/>
- [15] Smith William C and Etelson David J, "e-Business XML for Global Purchasing and the Supply Chain", International Federation of Purchasing and Materials Management (IFPMM) World Congress, South Africa 2001.
- [16] Sooriarachchi S.S., "A Graphical Tool for Management of ebXML Resources in Registry/Repository", B.Sc. Dissertation, University of Colombo School of Computing, May 2003.
- [17] Sun ebXML Registry and Repository Implementation, accessed on 071103, <http://www.ohelp.com/samples/xml/repreg/repreg-intro.html>
- [18] XML.ORG Goes Live with First Phase of Open Registry & Repository for XML Specifications, 2000, accessed on 071103, <http://lists.oasis-open.org/archives/announce/200006/msg00011.html>