Using UDDI to Find ebXML Reg/Reps

Registry Team

11 May 2001

(This document is the non-normative version formatted for printing, July 2001)

Table of Contents

1   Status of this Document

This document specifies an ebXML White Paper for the eBusiness community.

Distribution of this document is unlimited.

The document formatting is based on the Internet Society's Standard RFC format.

This version:

   http://www.ebxml.org/specs/rrUDDI.pdf

Latest version:

   http://www.ebxml.org/specs/rrUDDI.pdf

2   ebXML Participants

Sean Macroibeaird	Sun
Anne Thomas Manes	Sun
Scott Hinkelman	IBM
Barbara McKee	IBM

3   Introduction

The purpose of this document is to present a case study showing how to use the UDDI Business Registry to search for an ebXML Registry/Repository (Reg/Rep). This document , in accordance with the recommended use of UDDI, presents a series of steps that should be followed to define and register a Reg/Rep in a UDDI registry.

Sun and IBM submit this document for consideration within ebXML Registry specifications, and suggest that the document be referenced by those specifications.

4   Registering a Reg/Rep in UDDI

The steps shown here are deliberately broad and high-level as they represent a first-cut at this. Upon review extra detail and explanation can be added. If deemed appropriate, a representative of ebXML should register a number of canonical ebXML technical specifications in UDDI to facilitate this process for ebXML users.

4.1   Overview of the process

The data model specified in UDDI consists of four types of data entries. A businessEntity entry represents information about a business. Each businessEntity contains a set of businessService entries, which describe the services offered by the business. Each businessService entry contains a set of bindingTemplate entries, which provide pointers to the service access points. A bindingTemplate also refers to a tModel, which represents the technical specification of the service type.

In order to effectively use UDDI to search for a Reg/Rep, a tModel should be defined to represent Reg/Rep as a type of service. Each business that offers a Reg/Rep would register a Reg/Rep service, referring to the Reg/Rep tModel. This association will allow users to query the UDDI repository for Reg/Rep services.

4.2   Define the Reg/Rep tModel

A tModel represents a technical specification within UDDI. The format of the Reg/Rep tModel is as follows:

```
<tModel tModelKey="uuid:563726c8-bbb4-c5ec-768b-7d5c4fb690b7">

  <name>ebXML RegRep</name>

  <description lang="en">ebXML conformant
registry/repository</description>

  <overviewDoc>

    <description lang="en">EbXML Reg/Rep Specification</description>

<overviewURL>http://www.ebxml.org/project_teams/registry/private/Registry.dtd</overviewURL>

  </overviewDoc>

  <categoryBag>

    <keyedReference tModelKey="uuid:C1ACF26D-9672-4404-9D70-
39B756E62AB4" keyName="uddi: A specification "
keyValue="specification"/>
```

```
    <keyedReference tModelKey="uuid:C1ACF26D-9672-4404-9D70-
39B756E62AB4" keyName="uddi: An XML specification "
keyValue="xmlSpec"/>
    <keyedReference tModelKey="uuid:C1ACF26D-9672-4404-9D70-
39B756E62AB4" keyName="uddi: Using SOAP messages "
keyValue="soapSpec"/>
    </categoryBag>
  </tModel>
```

Notes:

- <tModelKey> : If supported as a Canonical tModel this will be defined in the UDDI v3.0 specifications. If not then it will be calculated by the operator when processing the **save_tModel** API call.

- <name>: This is a mandatory element and will be used to find registered tModels using the **find_tModel** API call.

- <description>: What is shown is a single English short description. It is possible to support other national language descriptions.

- <overviewDoc>: This allows a reference to a remote descriptive item related to the tModel. In the example above, the <overviewURL> element is the current URL for the Reg/Rep DTD file.

- <categoryBag>: When using the save_tModel it is possible to supply extra categorisation and identification information with the tModel. In this case the following keyedReferences are used, xmlSpec, specification, soapSpec[1].Create Business Registration for the Reg/Rep

In UDDI, all business services must be associated with a business entity. If a business wants to advertise that it hosts, owns, sponsors, or is represented by a Reg/Rep, that business must first create a businessEntity entry to represent its organization. The business registration process can be accomplished in multiple steps using the various API calls in the UDDI publisher API. In this example we show the entire process executed using a single **save_business** call. The BusinessServices offered by the Reg/Rep-owning organisation are defined within the businessEntity element. There may be many services defined, in addition to the service providing access to the Reg/Rep.

```
<save_business>
  <authInfo>secret</authInfo>
  <businessEntity businessKey="">
    <name>World-Wide Bicycle Parts Marketplace</name>
```

---

[1]As defined in Appendix I of UDDI API Programmers Specification.

```xml
    <categoryBag>

        <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Used bicyle (except motorised) shops "
keyValue="453310"/>

        <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Bicycles and parts manufacturing "
keyValue="336991"/>

        <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Bicycle rental " keyValue="532292"/>

        <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Bicycles (except motorised) wholesaling "
keyValue="421910"/>

    </categoryBag>

    <description lang="en">

      The World-Wide Marketplace for all bicycle parts

    </description>

    <contacts>

      <contact useType="president">

        <personName>Joe Murphy</personName>

        <description lang="en">President of A World-Wide Bicycle Part
Marketplace</description>

        <email useType="primary">joe.murphy@wwbmp.com</email>

        <address useType="http">

          <addressLine>http://www.wwbmp/contacts/</addressLine>

        </address>

      </contact>

    </contacts>

    <businessServices>
<!-- THE Reg/REP Business Service -->

      <businessService serviceKey="">

        <name>Search the World Bike parts Registry</name>

        <description lang="en">Get to the World Bicycle Parts
Registry</description>

        <bindingTemplates>

          <bindingTemplate bindingKey="">

            <accessPoint URLType="http">

              http://www.wwbmp.com/b2b/actions/search.jsp

            </accessPoint>

            <description lang="en">Use your Web Browser to search the
World Bike Parts registry</description>

            <tModelInstanceDetails>
```

```
                <tModelInstanceInfo tModelKey="uuid:563726c8-bbb4-c5ec-
768b-7d5c4fb690b7">

                    <description lang="en">ebXML RegRep</description>

                </tModelInstanceInfo>

            </tModelInstanceDetails>

          </bindingTemplate>

        </bindingTemplates>

      </businessService>

    </businessServices>

  </businessEntity>

</save_business>
```

Notes:

- businessEntity categoryBag: This element is used to classify the businessEntity. The businessEntity in the example is a bicycle parts consortium which hosts a Reg/Rep focused exclusively on this market. As such the businessEntity is classified as being a Reg/Rep, using the tModelKey for the Reg/Rep tModel, its market area is defined by referencing the NAICS code tModel with a key values of 336991, 421910, 532292, 453310 indicating various aspects of the world of bicycles and bicycle parts.

- businessServices: The service of interest here is the hosted Reg/Rep. The actual service provided is a searching service which enables the user to search the World-Wide Bicycle Parts Registry. Any further classifications which the modeler wishes to use to delineate the service can be entered in the businessService element definition.

- bindingTemplates: These are the exposed interfaces for registered businessServices. In the example this is an access point to a URL which will invoke a JSP search tool for the Reg/Rep. Conformance of the binding template to ebXML Reg/Rep is shown via the tModelInstanceDetails which refers to the ebXML Reg/Rep tModel.

5   Finding a Reg/Rep Using UDDI

5.1   *find_business*

There are several flavours possible using the **find_business** UDDI API call, i.e. find by business name, by identifierBag, by categoryBag, by tModelBag, by discoverURLs. For the purposes of this document the Reg/Rep will be located using the tModelBag as follows:

```
<find_business xmlns="urn:uddi-org:api" generic="1.0" maxRows="100">

  <tModelBag>

    <tModelKey>"uuid:563726c8-bbb4-c5ec-768b-7d5c4fb690b7"</tModelKey>

  </tModelBag>

</find_business>
```

This call searches the registry for businessEntities that have bindings that are compatible with a specific tModel pattern specified.

The return from a **find_business** call is a businessList containing businessInfo structures that match all the tModel keys passed:

```
<businessList xmlns="urn:uddi-org:api" generic="1.0"
operator="uddi.sourceOperator" truncated="false >

  <businessInfos>

      <businessInfo businessKey="....">

            <name>World-Wide Bicycle Parts Marketplace</name>

            <description lang="en">

                        The World-Wide Marketplace for all bicycle
parts

            </description>

            <serviceInfos>

                  <serviceInfo serviceKey="....">

                        <name>Search the World Bike parts
Registry</name>

                  </serviceInfo>

            </serviceInfos>

      </businessInfo>

  </businessInfos>
```

```
</businessList>
```

Notes:

- businessList: contains one or more businessInfo data sets. BusinessInfo are abbreviated version of the businessEntity data for use in further drill-down enquiries.

- businessInfo businessKey: The key value for a businessEntity calculated by the operator for the businessEntity when saved.

- serviceInfo serviceKey: The key value for a businessService calculated by the operator when the businessService was saved.

5.2   get_businessDetail

The next stage in the drill-down scenario is to retrieve all the information pertaining to a selected businessEntity. The **get_businessDetail** call is used to retrieve the businessServices and bindingTemplates associated with the selected businessEntity (via its businessKey).

```
<get_businessDetail xmlns="urn:uddi-org:api" generic="1.0">
    <businessKey> "....." </businessKey>
</ get_businessDetail>
```

Notes:

- businessKey : the businessKey from the selected businessInfo returned in the businessList.

A **get_businessDetail** call returns a businessDetail structure:

```
<businessDetail xmlns="urn:uddi-org:api" generic="1.0"
operator="uddi.sourceOperator" truncated="false">
  <businessEntity businessKey="…" authorizedName = "Fred"
operator="uddi.publishingOperator">
    <name>World-Wide Bicycle Parts Marketplace</name>
    <categoryBag>
      <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Used bicyle (except motorised) shops "
keyValue="453310"/>
      <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Bicycles and parts manufacturing "
keyValue="336991"/>
```

```
      <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Bicycle rental " keyValue="532292"/>

      <keyedReference tModelKey="uuid:C0B9FE13-179F-413D-8A5B-
5004DB8E5BB2" keyName="naics: Bicycles (except motorised) wholesaling "
keyValue="421910"/>

    </categoryBag>

    <description lang="en">

      The World-Wide Marketplace for all bicycle parts

    </description>

    <contacts>

      <contact useType="president">

        <personName>Joe Murphy</personName>

        <description lang="en">President of A World-Wide Bicycle Part
Marketplace</description>

        <email useType="primary">joe.murphy@wwbmp.com</email>

        <address useType="http">

          <addressLine>http://www.wwbmp/contacts/</addressLine>

        </address>

      </contact>

    </contacts>

    <businessServices>

      <businessService serviceKey=""…>

        <name>Search the World Bike parts Registry</name>

        <description lang="en">Get to the World Bicycle Parts
Registry</description>

        <bindingTemplates>

          <bindingTemplate bindingKey="">

            <accessPoint URLType="http">

              http://www.wwbmp.com/b2b/actions/search.jsp

            </accessPoint>

            <description lang="en">Use your Web Browser to search the
World Bike Parts registry</description>

            <tModelInstanceDetails>

              <tModelInstanceInfo tModelKey="uuid:563726c8-bbb4-c5ec-
768b-7d5c4fb690b7">

                <description lang="en">ebXML RegRep</description>

              </tModelInstanceInfo>

            </tModelInstanceDetails>

          </bindingTemplate>

        </bindingTemplates>
```

```
        </businessService>
      </businessServices>
    </businessEntity>
</businessDetail>
```

Notes:

- From the businessDetail information it is now possible to view the businessServices and bindingTemplates associated with Reg/Rep.

6   Ongoing Considerations

6.1   Canonical tModels

Given the relationship we wish to express between UDDI and ebXML, should the Reg/Rep tModel be supported as a Canonical tModel within UDDI? tModels for ebXML Core Components, CPPs, Business Process Definitions, etc. could also be supported by UDDI operators as Canonical tModels. This will prevent third parties outside UDDI and ebXML defining their own ad-hoc tModels.