Providing A Taxonomy For Use In UDDI Version 2
UDDI Working Draft Technical Note Document

July 17, 2001

This version:

*tn-taxonony-provider-V1.01-Final-20010717.pdf*

Latest version:

Authors:

Barbara McKee, IBM
David Ehnebuske, IBM

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION</td>
<td>3</td>
</tr>
<tr>
<td>PUBLISHING AN EXTERNAL TAXONOMY</td>
<td>3</td>
</tr>
<tr>
<td>To register an unchecked taxonomy</td>
<td>3</td>
</tr>
<tr>
<td>To register a checked taxonomy</td>
<td>4</td>
</tr>
<tr>
<td>Revocation</td>
<td>7</td>
</tr>
<tr>
<td>PERFORMING VALIDATION</td>
<td>7</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>8</td>
</tr>
</tbody>
</table>
Introduction

Taxonomies and identifier systems play an important role within UDDI. It is through categorization and identification that businesses are able to find each other and the services that meet their needs. Versions 1 and 2 of UDDI cite three common categorization schemes to encourage registrants to categorize their businesses, services and service descriptions. There are dozens of other taxonomies available that are newer, gaining in popularity, or targeted at specific constituencies. While UDDI does not mandate use of these taxonomies, it is imperative that they be made available to those who would benefit from using them.

A provider of a taxonomy or identifier system can allow unrestricted references to it or may choose to validate references. Taxonomy and identifier systems that allow unrestricted references are called unchecked. Conversely, a taxonomy or identifier system that requires references to it to be validated is called checked. A checked taxonomy or identifier system must have an associated validation service that performs value checking each time an attempt is made to save data containing a reference to the taxonomy or identifier system.

This paper guides the providers of taxonomies and identifier systems in the registration of their taxonomies and through the process of providing a validation service. Since taxonomies and identifier systems are handled in the same way, for conciseness this paper refers to both as “taxonomies”.

Publishing an external taxonomy

The first step in publishing a taxonomy is to decide whether it is to be checked or unchecked because the process of publishing them is different.

To register an unchecked taxonomy

To register an unchecked taxonomy you need only publish a tModel for it with a UDDI operator. Information on registering a tModel can be found in the UDDI Version 2 API Specification. For the name of your taxonomy, you should specify a urn that reveals its scope, its intent, or both. Use description to briefly describe what the taxonomy is used for. You may provide the description in multiple languages. You should make the overviewDoc point to a document describing the details of the taxonomy, including where to find (or how to compute) valid values and any restrictions on their use. You should classify the tModel for the taxonomy as being of types categorization and unchecked\(^1\) using the uddi-org:types taxonomy. If your tModel is for an unchecked identifier system rather than a categorization taxonomy, you should use identifier instead of categorization from the uddi-org:types taxonomy. Once you have published your taxonomy tModel in a UDDI registry it becomes immediately available for use in that registry. Your unchecked taxonomy tModel would look similar to this one:

```
<tModel authorizedName="..." operator="...
   tModelKey="uuid:1111111-2222-3333-4444-555555555555">
   <name>my-com:partner_types</name>
   <description xml:lang="en">
      Extendable taxonomy used to categorize my partners.
   </description>
```

\(^1\) The unchecked and checked categorizations are used to convey the intent of the provider. Neither is used by a UDDI registry to determine whether a validation service should be invoked to check referenced values or not. Instead, the presence of a validation service entry for the taxonomy or identifier system in one of the UDDI registry’s Operational Business Entities definitively declares the taxonomy or identifier system as checked. The absence of an associated validation service entry likewise asserts that the taxonomy or identifier system as unchecked.
To register a checked taxonomy

Registering a checked taxonomy is a six-step process that involves two parties: you -- the person or organization providing the taxonomy or identifier system, and the UDDI registry operator with whom you plan to publish the tModel for the taxonomy -- called the custodial operator. The six steps are:

1) You should begin the process by designing the tModel for your taxonomy. This is the same as what you would do for an unchecked taxonomy (see above) with three exceptions. First, you should mark the tModel with the checked categorization from the uddi-org:types taxonomy instead of the unchecked. Second, you must also classify the tModel with a categorization of unvalidatable from the uddi-org:types taxonomy. Marking the tModel unvalidatable prevents references to any values in the taxonomy or identifier system until every thing is ready to go. You will remove unvalidatable once the process is complete, making your taxonomy available for use. Third, you must not publish the tModel. You must instead contact a UDDI registry operator to do this, as described in the next step.

2) Next, you must choose a UDDI registry operator to act as the custodial operator and contact that operator, requesting that your tModel be published on your behalf. Once the tModel has been published, the custodial operator will own (i.e., have the ability to update) the tModel. Later when the validation service has been activated, the operator will transfer ownership to you, as described below. You will remove the unvalidatable categorization when you want to make the taxonomy available for categorization references.

Here is an example of an initial tModel for a checked taxonomy.

```xml
<overviewDoc>
  <description xml:lang="en">Initial list of types.</description>
  <overviewURL>
    http://www.my.com/partner_types.html
  </overviewURL>
</overviewDoc>

<categoryBag>
  <keyedReference>
    tModelKey="uuid:C1ACF26D-9672-4404-9D70-39B756E62AB4"
    keyName="uddi-org:types"
    keyValue="categorization"/>
  </keyedReference>
  <keyedReference>
    tModelKey="uuid:C1ACF26D-9672-4404-9D70-39B756E62AB4"
    keyName="uddi-org:types"
    keyValue="unchecked"/>
  </keyedReference>
</categoryBag>
</tModel>
```
3) You must then deploy a validation service for the taxonomy. This service must implement the `validate_values` message and return a disposition report indicating success or failure as described by the `uddi-org:taxonomy_V2 tModel`. Details about the functions this service is required to perform can be found in Appendix H of the UDDI Version 2 API Specification.

4) The custodial operator assesses the validation service. While each operator has its own set of tasks that it performs during the assessment, you can expect some testing of the service to ensure that it performs at an acceptable level. The custodial operator will also require certain pieces of information to enable it to assess the validation service. These are:

- Whether caching of valid values is allowed by UDDI registry operators. If caching is allowed, operators may save data that refers to the taxonomy or identifier systems and avoid continually calling the validation service if the validation service has previously checked and okayed the values involved. You should permit caching if you can. Permitting caching improves the performance of the publishing operation the operator is handling, and reduces the load on the validation service. You should not permit caching if your validation service performs contextual examination of the entity being published or if the operator cache can become invalid.

- A `bindingTemplate` for the validation service. The contents of this `bindingTemplate` will be used by the custodial operator to set up the actual service binding for your validation service. Because UDDI registries have performance standards to maintain in calling out to validation services such as yours, UDDI registry operators require that the `accessPoint URL` in the `bindingTemplate` not be published elsewhere in the same UDDI registry. The `bindingTemplate` should reference the `uddi-org:taxonomy_V2 tModel`, which has the `tModelKey` `uuid:1E3E9CBC-F8CE-41AB-8F99-88326BAD324A`. 
5) The custodial operator registers your validation service in its Operational businessEntity. When the custodial operator successfully completes the assessment of your validation service, it publishes a bindingTemplate for the validation service within its own Operational businessEntity. The bindingTemplate will use a hostingRedirect, as described in Appendix G of the UDDI Version 2.0 API Specification, instead of the accessPoint you provided to conceal the actual location of the service. This reduces the risk of denial of service attacks. If at any time in the future the accessPoint of the validation service changes, you must notify the custodial operator of the new accessPoint since it is through the operator’s redirected service binding that the validation service is actually invoked.

What follows is an example of an approved checked taxonomy service binding in a custodial node operator’s Operational Business Entity:

Here is an example of a binding template for the validation service that is submitted to the custodial operator:

```xml
<bindingTemplate bindingKey="">
  <description xml:lang="en">Validation service binding for my-com:realtor_types</description>
  <accessPoint URLType="http">
    http://www.realtor.com/realtor_validation
  </accessPoint>
  <tModelInstanceDetails>
    <tModelInstanceInfo>
      tModelKey="uuid:22222222-3333-4444-5555-6666666666666666"
    </tModelInstanceInfo>
    <tModelInstanceInfo>
      tModelKey="uuid:1E3E9CBC-F8CE-41AB-8F99-88326BAD324A"
    </tModelInstanceInfo>
  </tModelInstanceDetails>
</bindingTemplate>
```

```xml
<businessService businessKey="..." serviceKey="...">
  <name>Approved validation services</name>
  <description>Set of service bindings for approved validation services</description>
  <bindingTemplates>
    <bindingTemplate bindingKey="..." serviceKey="...">
      <description xml:lang="en">Validation service binding for my-com:realtor_types</description>
      <hostingRedirect bindingKey="... of redirector service binding ...">
      </hostingRedirect>
      <tModelInstanceDetails>
        <tModelInstanceInfo>
          tModelKey="uuid:22222222-3333-4444-5555-6666666666666666"
        </tModelInstanceInfo>
        <tModelInstanceInfo>
          tModelKey="...tModelKey for uddi-org:taxonomy_V2..."
        </tModelInstanceInfo>
      </tModelInstanceDetails>
    </bindingTemplate>
  </bindingTemplates>
</businessService>
```
Once the custodial operator creates the service binding and has set up the hosting redirector, it will transfer ownership of the tModel to you and notify you that everything is ready to go.

6) You should then remove the unvalidatable categorization from the tModel, thus causing the taxonomy to become active.

If at any time you wish to disable the validation service, you may re-apply the unvalidatable categorization to the taxonomy’s tModel. Attempts to categorize using the taxonomy will then fail until the unvalidatable categorization is once again removed. However, existing references to a taxonomy that has been marked unvalidatable are “grandfathered” such that updates to the referencing entity will not fail due to the unavailability of the validation service, but new references will not be able to be validated and will be rejected. Since disabling an existing taxonomy can be quite inconvenient for its users, you should avoid it when possible.

**Revocation**

Each UDDI registry operator has policies for handling validation services that fail to perform acceptably. These range from making the taxonomy or identifier system temporarily unavailable, to removing the approved service. If your registered validation service fails to operate acceptably, the custodial operator will likely contact you to work towards finding a remedy to the problem. First steps will often include taking the taxonomy or identifier system out of service by adding the unvalidatable categorization back to the taxonomy’s tModel. When the problem is resolved, the unvalidatable categorization is again removed. If you cannot get the validation service back on line, the operator can remove your validation service from its Operational businessEntity. This has the affect of making the taxonomy or identifier system unchecked. An unchecked taxonomy that is in use cannot be subsequently converted (back) to a checked taxonomy.

**Performing validation**

The validation service for a checked taxonomy or identifier system is invoked by the registry operator to validate attempts to publish entities that reference that taxonomy. When an entity being published contains at least one keyedReference within it that references the tModelKey of the checked taxonomy or identifier system, the taxonomy’s associated validation service is invoked (unless the values specified in the entity are cached). The service is passed the entity being published. The validation service must inspect each relevant keyedReference in the entity for invalid values. The service can merely check that the values are from an allowed set, or it may perform contextual validation using the entity information it is passed.

Because UDDI operators call validation services during a publish operation, they are encouraged to optimize their validation service invocations by calling only once the validation services that share a common access point. Thus, if a validation service is capable of validating several taxonomies or identifier systems, and the validation service bindings for each of the taxonomies and identifier systems have the same access point, the validation service must validate in a single invocation all the references from all the taxonomies and identifier systems that it knows about in the entity passed.

A validation service must:

1) Check the validity of keyValues in all applicable keyedReferences it is passed.

2) Return the result of the validation in a dispositionReport. If all applicable keyValues are valid, **success** is returned in the dispositionReport. If validation fails for any applicable keyValue, an
error message explaining why at least one keyValue failed validation should be captured in the errInfo element content and returned in a disposition report with the E_invalidValue or E_valueNotAllowed error. The validation service may stop after the first invalid keyValue is detected, or it may check all of the applicable values.

References