

Importing new canonical tModels: A utility for updating your WebSphere Application Server UDDI registry

Skill Level: Intermediate

[Bernard Z. Kufluk \(bernard@uk.ibm.com\)](mailto:bernard@uk.ibm.com)

Software Engineer
IBM

[Ian Hodges \(ihodges@uk.ibm.com\)](mailto:ihodges@uk.ibm.com)

Software Engineer
IBM

29 Apr 2009

Keep your Universal Description Discovery and Integration (UDDI) registry current with this simple utility for importing new canonical tModels (technical models). The utility ensures that newly defined canonical tModels are saved to your IBM® WebSphere® Application Server UDDI registry database with defined keys for both V2 and V3.

Introduction

This article introduces a useful utility for importing canonical tModels (short for technical models) into your existing WebSphere Application Server V6 UDDI Registry. When a new UDDI Registry node is created it is populated with all UDDI canonical tModels, sanctioned by the Organization for Advancement of Structured Information Standards (OASIS) technical committee and known prior to the release of a WebSphere version. Over time, new tModels are adopted by the technical committee and become canonical tModels. These tModels are often produced as a result of new standards. If you wish to reference any new canonical tModels in your UDDI Registry and keep up with emerging standards, then you can use this utility to import these into your UDDI database.

About tModels

A tModel is a data structure representing a reusable concept, such as a Web service type, a protocol used by Web services, or a category system. tModel keys in a service description are a technical "fingerprint" that you can use to trace the compatibility origins of a given service. They provide a common point of reference so that you can identify compatible services.

Canonical tModels represent a well-known concept and are owned by a well-known standards group, an industry vertical, or a consortium. See [Resources](#) for a listing of tModels documented by OASIS.

One difference between canonical tModels and ordinary tModels is that canonical tModels have two well-known keys for each tModel; the V2 specification key and the V3 specification key. Thus, in any UDDI Registry the tModel can be referenced by this key when using either the V2 or V3 SOAP API. Using the V3 save_tModel SOAP API a tModel can be saved to a UDDI Registry with a defined V3 key. However, the V2 key cannot be defined, and the node will assign the key as per the V2 specification. With this utility, you can save the canonical tModels with both V2 and V3 keys defined.

Download the utility and configure the properties file

Table 1 lists the contents of the [UDDICanonicalTModelImportTool.zip](#) file.

Table 1. The contents of UDDICanonicalTModelImportTool.zip

UDDICanonTModelImportTool.jar
UDDICanonicalTModelImportTool.properties
WSRP-TModels.xml
PolicyModels.xml

After downloading the utility, use the supplied properties file to specify connection information for the UDDI registry database and the location of message and trace files. For each supported database, the properties file contains a section for specifying connection details. For your database type, remove the comment character '#' from each of the properties. Ensure that only one database type is active. Listing 1 shows an example of DB2 connection details.

Listing 1. UDDICanonicalTModelImportTool.properties database configuration file

```
#####
# Configuration for destination UDDI DB (DB2) #
# jar DB2_HOME/java/db2java.zip                #
```

```
#####
dbDriver=COM.ibm.db2.jdbc.app.DB2Driver
dbUrl=jdbc:db2:uddi30
dbUser=db2admin
dbPasswd=db2password
dbDataSchema=IBMUDI30
dbSystemSchema=IBMUDS30
```

The comment header denotes the database type and the location of the JAR files containing the JDBC driver. `dbDriver` names the JDBC driver to load to enable the database connection, `dbDataSchema` and `dbSystemSchema` denote the schema name used to access the UDDI Registry database tables. These properties should not be changed.

`dbUrl` is used to denote the name or the location of the UDDI Registry database dependent on the particular RDBMS. The `dbUser` and `dbPasswd` are used to define the userid and password of a user with permission to insert into the UDDI Registry database.

If your RDBMS type is embedded IBM® Cloudscape™ (WebSphere Application Server V6) or embedded Apache Derby (WebSphere Application Server V6.1) then the database can only be accessed by one JVM at a time. You must first stop the application server hosting the UDDI registry to allow the import utility to access the database.

The WSRP-TModels.xml file is an example file for testing the utility. It contains a number of canonical tModels defined by the Web Services for Remote Portlets (WSRP) specification ([see Resources](#)). These tModels are defined using the normal UDDI standard, but the `tModel` element contains the attributes `v2tModelKey` and `v3tModelKey` (instead of the attribute `tModelKey`). For example:

```
<tModel
v2tModelKey="uuid:58a98609-c265-3c28-9079-85ea8b2521ef"
v3tModelKey="uddi:oasis-open.org:wsrp:service_type">
```

For your canonical tModels create an XML file with a `canonicalTModels` element containing all the tModel definitions as described by an appropriate document. For each tModel define the V2 and V3 keys.

The utility reads and validates the XML file, connects to the database defined in the properties file, and inserts the canonical tModels into the UDDI Registry database. An option also exists to delete canonical tModels defined in the XML file from the UDDI Registry database.

Run the utility

Once the properties file and the XML files have been set up, you can run the utility. The classpath must include the JAR files containing your JDBC driver. See the

properties file for details of usual locations. Listing 2 shows the syntax for the utility. Substitute `./jdbcDriver.jar` with the path of your JDBC driver files and *filename* with the file containing the canonical TModels. The `-u` option removes the canonical tModels previously imported.

Listing 2. Syntax for the tModel import utility

```
java -classpath
./jdbcDriver.jar;{WAS_HOME}/lib/uddi4jv2.jar;./UDDICanonTModelImportTool.jar
com.ibm.uddi.v3.ctmit.CanonicalTModelImportTool -f <filename> [-u]
```

The following example commands illustrate importing and deleting tModels from the registry database.

```
java -classpath ./derby.jar;/AppServer/lib/uddi4jv2.jar;./UDDICanonTModelImportTool.jar
com.ibm.uddi.v3.ctmit.CanonicalTModelImportTool -f tmodel.xml

java -classpath ./derby.jar;/AppServer/lib/uddi4jv2.jar;./UDDICanonTModelImportTool.jar
com.ibm.uddi.v3.ctmit.CanonicalTModelImportTool -f tmodel.xml -u
```

Listing 3 contains example output. If there is an error in the database connection information or in the format of the tModels defined in the XML, these errors are reported.

Listing 3. Example output of the import utility

```
CWUDU5003I: Importing entities...
CWUDU5006I: Deserializing...
CWUDU5007I: Deserialized entities.
CWUDU5005I: Imported tModel, tModelKey[uddi:oasis-open.org:wsrp:service_type].
```

If there is an error in the tModel description or you no longer require the tModels, re-run the utility with the `-u` option to remove the tModels from the database.

Verify your registry content

Once the tModels have been imported, use the UDDI User Console to search for these tModels and verify that they are described correctly in the UDDI Registry, as shown in Figure 1.

Figure 1. WSRP tModels loaded into UDDI

Registered Technical Models: 8	
Names	
WSRP_v1_Markup_Binding_SOAP	
WSRP_v1_Markup_PortType	
WSRP_v1_PortletManagement_Binding_SOAP	
WSRP_v1_PortletManagement_PortType	
WSRP_v1_Registration_Binding_SOAP	
WSRP_v1_Registration_PortType	
WSRP_v1_ServiceDescription_Binding_SOAP	
WSRP_v1_ServiceDescription_PortType	

The canonical tModels are now available for referencing from other UDDI entities.

Summary

This article describes a utility for importing new canonical tModels into an Application Server UDDI registry database. It allows you to save tModels with keys defined for both UDDI V2 and V3.

Downloads

Description	Name	Size	Download method
Canonical tModel import tool and sample data	UDDICanonicalTModelImportTool.zip	216KB	HTTP

[Information about download methods](#)

Resources

- Read more on the [Organization for the Advancement of Structured Information Standards \(OASIS\)](#) standards for Canonical tModels.
- Read the [Web Services for Remote Portlets \(WSRP\)](#) specification at OASIS.
- The [Web Services Policy 1.5 - Attachment Specification](#) includes the specification of the Policy tModels in an appendix.

About the authors

Bernard Z. Kufluk

Bernard Kufluk is an IBM Software Developer working on WebSphere Service Registry and Repository. He has been engaged in the API development team since November 2005. Over the past seven years, he has worked on various IBM products including WebSphere Voice Application Access, WebSphere Voice Response, and Intelligent Notification Services. You can reach him at bernard@uk.ibm.com.

Ian Hodges

Ian Hodges is an IBM Advisory Software Engineer at IBM Hursley Laboratories, UK, developing Web Technologies. He was previously a lead developer of the WebSphere UDDI Registry. Ian has worked with Java technologies for 10 years and is a Sun Certified Java Developer. He has over 20 years of software development experience. You can reach him at ihodges@uk.ibm.com.

Trademarks