Managing WebSphere Process Server Relationship Service database objects

Understanding when and how the system creates and deletes database objects

Olga Tikhonova

November 12, 2008

This article explains how the Relationship Service in IBM WebSphere Process Server handles the creation and the deletion of the database schema objects that it maintains for every relationship installed on WebSphere Process Server. It clarifies why and when the system deletes objects. In this article, you use a convenient utility to return the list of database object names for a given relationship or delete the objects from the database. Prior knowledge of WebSphere Process Server Relationship Service is recommended.

Introduction

Relationship Service maintains a set of database schema objects to hold and manage the relationship instance data. The objects include tables, stored procedures, and sequences (and views starting with IBM® WebSphere® Process Server V6.2). They reside in the relationship database. You can configure database access during the installation of WebSphere Process Server. If needed, you can modify it at a later point using Relationship Manager in the administrative console, see Figure 1 (link to enlarged Figure 1).

Figure 1. Accessing Relationship Service configuration in the Relationship Manager tool in the administrative console
Figure 2. Configuring the data source for the Relationship database

Tables store the relationship instance data. The Relationship Service runtime uses stored procedures and sequences to add/delete/update the entries in the tables.

During application deployment, the system creates a set of database objects in the relationship database for each relationship definition that the application contains, see Figure 2. The object names are generated dynamically using the combination of the relationship and role names and the generated unique identifier. The tables contain the "mission critical" relationship instance data, so the only recommended way to modify this data is through the Relationship Manager application in the administrative console.

When the application is undeployed from the server, whether the database objects are preserved or deleted from the database, depends on which profile the server uses. Two server profiles determine the difference in the behavior of the Relationship Service: the integrated test client profile and the standalone server profile.

The rest of this article explains why and when the relationship database objects can and cannot be deleted.

Integrated test client (UTC) profile

If an application containing some relationship is deployed into the UTC, and you choose to undeploy it, then Relationship Service deletes all database objects that correspond to the deleted relationship and data that they contain from the relationship database. In this scenario, it is assumed that the application user is in design mode and it is considered safe to delete the relationship instance data and the schema objects. In this way, if you modify the relationship definition and redeploy the application into the UTC, no inconsistencies between the new definition and the old schema objects are possible. The downside of this assumption is that any data that the old database objects might have contained is lost when the application is undeployed from the UTC. The common known workaround for this limitation is to back up the data prior to application undeployment and subsequent copying it back into the tables after the application is redeployed.
Of course the direct copy is only possible if the old and new object schemas are compatible. If they are not compatible, for example, the key attribute is changed or added in the role definition, then copying the backed up instance data into the new tables might not be possible or is harder to achieve. Moreover, if the relationship definition is altered, the old instance data might not even be applicable any longer and can be discarded.

**Standalone WebSphere Process Server profile**

If an application containing a relationship definition is deployed and undeployed on/from WebSphere Process Server, then the Relationship Service only removes the relationship definition from its memory, but preserves all database objects for it. This scenario assumes that the application might be used on the production system and the relationship tables contain "mission critical" runtime data that is not safe to remove automatically. Another consideration is that the relationship data can be in-use by other applications. The downside of these assumptions is if the user updates the relationship in such a manner that makes it incompatible with the old database schemas and redeployed the application, then the Relationship Service will not be able to install the updated relationship correctly. In such a case the Relationship Service writes the error message into the log and disables the relationship.

If the new table schema is different from the existing one, then Relationship Service throws the following exception: "Table table name already exists, but the table schema is different from the current role".

If the new stored procedure parameters do not match the existing ones, then Relationship Service throws exception and logs the following error message: "CWLAE0007E: The stored procedure stored procedure name already exist, but its parameters are inconsistent with the current role definition".

The only way out of such scenario is to drop all database objects for that relationship manually, for example, via the database GUI, and restarting the server. During server reboot, the database objects are recreated according to the updated relationship definition. This solution method is complicated by the fact that the database objects have cryptic names, and it can be difficult and time consuming to figure out the right and complete set of objects to delete. It is also possible to delete the wrong object, for example, table or stored procedure for the different relationship with a similar name, but completely unrelated. This obviously causes problems for that second relationship.

This article provides a utility to assist those who find themselves in the latter situation. Based on the input parameter, you can use the utility to either return the names of all database objects for a given particular relationship, or delete the database objects associated with a given relationship from the database. Download and unzip the zip file in the Download section of this article. Follow the instructions in the readme.txt file to use the utility.

**Conclusion**

This article explained how the Relationship Service maintains the database objects used for keeping the runtime instance data. The article clarified how the database objects are handled
during application uninstall. If you find yourself in a situation where you updated your relationship definition, but ran into the schema incompatibilities, then you can use the utility below to drop all database objects for your relationship. If you prefer to manually delete the objects, the utility can find all the database objects for your relationship and you can then drop them manually using database GUI. Either way, after the objects are deleted, you need to reboot WebSphere Process Server. At this point, the new database objects will be created by the Relationship Service runtime based on the updated relationship definition.
# Downloadable resources

<table>
<thead>
<tr>
<th>Description</th>
<th>Name</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utility program for locating relationship schemas</td>
<td>RelSchemaFinder.zip</td>
<td>10KB</td>
</tr>
</tbody>
</table>

© Copyright IBM Corporation 2008  
Trademarks  
(www.ibm.com/developerworks/ibm/trademarks/)