How to ... Configure UD Connect on the J2EE Server for JDBC Access to External Databases

Version 1.05 – Jan. 2004

Applicable Releases:
SAP NetWeaver '04 (SAP BW3.5)
1 Scenario

You have data in a non-SAP database that you wish to report against using the new Universal Data Connect (UDC) functionality introduced in SAP BW 3.5. By using a remote cube to access the external data, you can create queries, use the Bex Analyzer and create web applications just like you would for SAP BW data.

2 Introduction

UD Connect utilizes the J2EE engine of the SAP Web Application server to access external data. This How-to will deal with the set-up and configuration of the relational JDBC connector. Connectors to access multi-dimensional OLAP databases are also available, but not covered in this guide.

This guide will detail the steps necessary to connect the SAP BW ABAP stack with the J2EE stack via RFC as well as configuring the JDBC connector to use the database vendor provided JDBC driver. It is assumed you have installed the J2EE server and the Universal Data Integration (UDI) specific add-ons from the installation DVD.
3 The Step By Step Solution

1. Start the J2EE 'Visual Administrator' via the `go.bat` file located here:
   `<drive>\usr\sap\<SID>\DVEBMGS00\j2ee\admin`.
   
   Note: Creating a shortcut to 'go.bat' and placing it on the desktop will make future access more convenient.

2. Logon to the J2EE server. After you are authenticated, you should see a screen similar to the one on the right.
3. Expand the 'Server' node and the 'Services' node and click on the 'JCo RFC Provider' service.

4. You will add a RFC server that upon startup, logs into your SAP BW 3.5 system and registers itself. The field 'Program ID' is freely definable, but write it down, you will need it later. Complete the fields. 'User' is a 'system' user, similar to the ALE_REMOTE user. Once all the fields are filled in, click on 'Set'. The RFC server will automatically start.

Note: If your BW system is Unicode, make sure the 'Unicode' field is checked, otherwise leave it unchecked.
5. Defining a RFC destination in BW. Create a new RFC destination of type 'TCP/IP' via transaction SM59.

Click on the 'Technical Settings' tab and select the 'Registered Server Program' radio button. Enter the name of the RFC server from the previous step in the field 'Program ID'. In this example – 'rgphl101_j2ee'.

Test your connection.

6. Back in in the J2EE Visual Administrator, select the 'JDBC Connector' Service and expand the 'Drivers' node.

Click on the sheet-of-paper icon to create a new entry.

7. In the dialog box, enter a freely definable name and click 'OK'. Specify the location of the JDBC driver JARs provided to you by your RDBMS vendor. Keep adding JARs until you are done.

Note: The JDBC driver for Microsoft SQL Server is available from Microsoft at http://www.microsoft.com/downloads/details.aspx?FamilyID=9f1874b6-f8e1-4bd6-947c-0fc5b05b71&DisplayLang=en
When you are done adding JARs you should have an entry for your new driver.

8. Navigate to the ‘Connector Container’ Service and select the ‘BI_JDBC_CONNECTOR’.

   We need to add a load reference that starts with ‘library’.

   IMPORTANT: The string after the colon must match the name you used in step 7, where you defined the DB Driver!
   Example: library:MSSQL
10. Navigate to the ‘Connector Container’ service, click on your connector and then on the right hand side of the screen click on the ‘Managed Connection Factory’ and then the ‘Properties’ tab.

11. Click on the ‘DriverName’ key row and change the value to the string specified in your JDBC documentation. Repeat for the ‘URL’ key. Enter the database username and password used for logging into the DB.

Example (MS SQLServer only!):
DriverName: com.microsoft.jdbc.sqlserver.SQLServerDriver
URL: jdbc:microsoft:sqlserver://<server>:<port>;database=Northwind

Notes:
<server> is the complete hostname (e.g. myhost.acme.com)
The default port for MS SQLServer is 1433.
Important: Click on the ‘Save’ icon when you are done with the configuration.
12. To perform a quick ‘smoke test’, execute the following URL in your browser, replacing the placeholders with your server name and J2EE port.

http://<server>:<port>/TJdbc/servlet/Test

13. You have successfully configured the J2EE server for UD Connect! At this point you can clone the BI_JDBC_CONNECTOR in order to provide more granular access to your database (i.e. using different users and schemas).

Note: You must start cloned connector names with 'SDK_' in order for them to be visible in the BW backend system for selection during the mapping process.