

## **Oracle® Fusion Middleware**

Getting Started with SAP ABAP ERP Adapter for Oracle Data  
Integrator

11g Release 1 (11.1.1)

**E39326-02**

August 2013

Oracle Fusion Middleware Getting Started with SAP ABAP ERP Adapter for Oracle Data Integrator, 11g Release 1 (11.1.1)

E39326-02

Copyright © 2011, 2013, Oracle and/or its affiliates. All rights reserved.

Primary Authors: Laura Hofman, Joshua Stanley

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

---

---

# Contents

|   |     |
|---|-----|
| <b>Preface</b> .....  | v   |
| Audience .....  | v   |
| Documentation Accessibility .....                                     | v   |
| Related Documents .....   | v   |
| Conventions .....   | vi  |
| <br>  |     |
| <b>1 Overview of Oracle Data Integrator SAP ERP Knowledge Modules</b> |     |
| Overview .....  | 1-1 |
| <br>  |     |
| <b>2 Setting Up the Environment</b>                                   |     |
| Before You Begin .....  | 2-1 |
| System Requirements and Certification .....                           | 2-1 |
| Getting the Right Privileges .....                                    | 2-2 |
| Gathering SAP Connection Information .....                            | 2-3 |
| Validating the Shared Folder Setup .....                              | 2-3 |
| Validating the FTP Setup .....  | 2-4 |
| Validating SAP Privileges .....                                       | 2-5 |
| Validating SAP Transport Layer Name .....                             | 2-5 |
| Installing Oracle Data Integrator .....                               | 2-6 |
| Installing and Configuring the Oracle DB Target .....                 | 2-6 |
| Installing and Configuring JCo .....                                  | 2-7 |
| Setting Up an FTP Server .....  | 2-8 |
| Configuring Oracle Data Integrator .....                              | 2-8 |
| Set Up the Topology .....   | 2-8 |
| Add the Open Tool .....   | 2-9 |
| <br>  |     |
| <b>3 Configuring the Topology</b>                                     |     |
| Configuring the Target Data Server .....                              | 3-1 |
| Configuring the SAP Source Server .....                               | 3-4 |
| Configuring the Data Server .....                                     | 3-4 |
| Configuring the Logical Schema .....                                  | 3-6 |
| <br>  |     |
| <b>4 Creating a New Project</b>                                       |     |
| Creating a New Project .....  | 4-1 |

## 5 Reverse-Engineering Data Models

|   |     |
|---|-----|
| Reverse-Engineering the Oracle Target.....          | 5-1 |
| Reverse-Engineering the SAP Source Datastores ..... | 5-2 |
| Creating the Data Model .....                       | 5-3 |
| Validating the SAP Connection .....                 | 5-3 |
| Starting the Reverse-Engineering Process .....      | 5-5 |
| Using the SAP Metadata Browser .....                | 5-7 |
| The Tree View .....                                 | 5-8 |
| The Search Panel .....                              | 5-9 |
| Reverse-Engineering Without GUI .....               | 5-9 |

## 6 Creating and Running the Integration Interface

|   |     |
|---|-----|
| Creating the Integration Interface.....       | 6-1 |
| Create the Interface .....                    | 6-1 |
| Define the Source and Target Datastores ..... | 6-3 |
| Define Joins between Sources .....            | 6-4 |
| Create the Mappings .....                     | 6-5 |
| Define the Interface Flow .....               | 6-6 |
| Running the Integration Interface .....       | 6-7 |
| Running the Interface .....                   | 6-8 |
| Review the Interface Execution .....          | 6-8 |
| Review the Resulting Data .....               | 6-9 |

## 7 Going Further with Oracle Data Integrator

|   |     |
|---|-----|
| Going further with Oracle Data Integrator ..... | 7-1 |
|---|-----|

### A SAP ABAP ERP Required Privileges

|   |     |
|---|-----|
| Authorizations Required for RKM SAP ERP Setup .....           | A-1 |
| Authorizations Required for RKM Execution .....               | A-3 |
| Authorizations Required for LKM Execution .....               | A-3 |
| Authorizations Required for LKM Execution for Production..... | A-3 |

### B Installing the librfc32/librfc64 Library

|   |     |
|---|-----|
| Installing the librfc32/librfc64 Library..... | B-1 |
|---|-----|

### C SAP Stand-Alone Connection Test

|                                       |     |
|---------------------------------------|-----|
| SAP Stand-Alone Connection Test ..... | C-1 |
|---------------------------------------|-----|

### D Uninstalling ODI SAP Components

|                                       |     |
|---------------------------------------|-----|
| Uninstalling ODI SAP Components ..... | D-1 |
|---------------------------------------|-----|

---

---

# Preface

The purpose of this document is to provide you with an overview of the possibilities offered by SAP ERP Knowledge Modules and how to use them in Oracle Data Integrator.

This document provides examples to help you understand how to achieve an ETL project with Oracle Data Integrator, loading a Data Warehouse with data from SAP systems, while validating data quality.

This project should take approximately 90 minutes to complete. You may save your work in order to pause and resume at any point in the project.

It is recommended that you use Oracle Data Integrator before proceeding with this guide. A good introduction to Oracle Data Integrator is the guide *Oracle Fusion Middleware Getting Started with Oracle Data Integrator*.

It is recommended that you also review the SAP ABAP chapter of the *Oracle Fusion Middleware Knowledge Module Developer's Guide for Oracle Data Integrator* for more information about the SAP ERP KMs.

## Audience

This document is intended for developers and administrators who want to use Oracle Data Integrator Application Adapters with SAP ERP systems using SAP JCo libraries, as a development tool for their integration processes.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Related Documents

For more information, see the following Oracle resources:

- *Oracle Fusion Middleware Getting Started with Oracle Data Integrator*
- *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator*
- *Oracle Fusion Middleware Installation Guide for Oracle Data Integrator*
- *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*
- *Oracle Fusion Middleware Connectivity and Knowledge Modules Guide for Oracle Data Integrator*
- *Oracle Fusion Middleware Knowledge Module Developer's Guide for Oracle Data Integrator*
- *Oracle Fusion Middleware Getting Started with SAP ABAP BW Adapter*
- *Oracle Data Integrator 11g Online Help*
- *Oracle Data Integrator 11g Release Notes*, included with your Oracle Data Integrator 11g installation and on Oracle Technology Network

## Conventions

The following text conventions are used in this document:

| <b>Convention</b> | <b>Meaning</b>   |
|-------------------|--|
| <b>boldface</b>   | Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.         |
| <i>italic</i>     | Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.                          |
| monospace         | Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter. |

---

---

# Overview of Oracle Data Integrator SAP ERP Knowledge Modules

This chapter provides an overview of Oracle Data Integrator SAP ERP Knowledge Modules.

## Overview

The Oracle Data Integrator SAP ABAP Knowledge Modules provide integration from SAP ERP systems using SAP JCo libraries. This set of KMs has the following features:

- Reads SAP data from SAP ERP system
- Loads this SAP data into Oracle or non-Oracle Staging Area
- Reverse-engineers SAP metadata and proposes a tree browser to pick up only the required metadata

This adapter includes three knowledge modules:

- **RKM SAP ERP:** Its main role is to perform customized reverse-engineering of SAP tables into ODI datastores (Models). The SAP ERP RKM is in charge of connecting to the SAP System, which provides the SAP table's metadata information, and transforming and writing the resulting metadata into Oracle Data Integrator's repository.
- The **LKM SAP ERP to Oracle (SQLDR)** connects to SAP and retrieves data from the SAP tables defined in the SAP Model. It is used in the interfaces. The LKM is in charge of loading source data from a SAP System to an Oracle staging area.
- The **LKM SAP ERP to SQL** connects to SAP and retrieves data from the SAP tables defined in the SAP Model. It is used in the interfaces. The LKM is in charge of loading source data from a SAP System to a non-Oracle staging area.



---

---

## Setting Up the Environment

This chapter describes how to set up your SAP ERP environment for use with Oracle Data Integrator.

It includes the following topics:

- [Before You Begin](#)
- [Validating the Shared Folder Setup](#)
- [Validating the FTP Setup](#)
- [Validating SAP Privileges](#)
- [Validating SAP Transport Layer Name](#)
- [Installing Oracle Data Integrator](#)
- [Installing and Configuring the Oracle DB Target](#)
- [Installing and Configuring JCo](#)
- [Setting Up an FTP Server](#)
- [Configuring Oracle Data Integrator](#)

### Before You Begin

Before you begin, consider the following:

- [System Requirements and Certification](#)
- [Getting the Right Privileges](#)
- [Gathering SAP Connection Information](#)

### System Requirements and Certification

Before performing any installation you should read the system requirements and certification documentation to ensure that your environment meets the minimum installation requirements for the products you are installing.

The Oracle Data Integrator requirements are listed in the *Oracle Fusion Middleware Installation Guide for Oracle Data Integrator*.

The requirements specific to the Oracle Data Integrator SAP ABAP Adapter are:

- Oracle Data Integrator 11.1.1.7.0 or above.

---

---

**Note:** Most of this documentation also applies to the SAP connectivity in Oracle BI-Applications, which requires Oracle Data Integrator 10g.

---

---

- A Java Connector (JCo) version compatible with adapter must be used. The list of supported JCo versions is available in the Certification Matrix available from Oracle Technology Network (OTN). A minimum version of JCo 3.0.2 is required.
- A JVM version compatible with both Oracle Data Integrator and JCo must be used. A minimum version of JVM 1.5 is required due to JCo pre-requisites.
- The connector supports two transfer modes for transferring data from SAP system to the ODI agent: data transfer using a Shared Directory and data transfer through FTP. For details and restrictions see Section 8.7.1, "File Transfer Considerations" of *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator*. Depending on the chosen file transfer mode the following requirements must be met:
  - **Data transfer through a Shared Directory (recommended transfer method)**  
The LKM SAP ERP to Oracle (SQLLDR) requires a folder that is shared between the SAP system and the ODI agent. The SAP application server transfers the data by writing it out into a folder that is accessible from the SAP system and the ODI agent machine. This is typically done by sharing a folder of the ODI agent machine with the SAP system. Note that the shared folder does not necessarily have to be located on the ODI agent machine. A shared folder on a third machine is also possible, as long as the shared folder is accessible to both the ODI agent machine and the SAP system.

---

---

**Note:** For security reasons, the SAP folders should not be shared with the ODI agent. Share only folders of the ODI agent machine with the SAP system.

---

---

The shared folder must be accessible to SAP system and not just to the underlying operating system. This means that the folder needs to be declared in SAP transaction AL11 and the folder opens successfully in AL11.

- **Data transfer through FTP** LKM SAP ERP to Oracle (SQLLDR) requires a FTP server to upload data from the SAP ERP system. This data is either read locally by the agent executing the interface (when this agent runs on the FTP server machine), or remotely (when this agent is located on a different machine than the FTP server). This FTP server must be accessible over the network from both the SAP ERP machine and the agent machine.
- For LKM SAP ERP to Oracle (SQLLDR) only: SQL\*Loader is required on the machine running the agent the executed interfaces using LKM SAP ERP to Oracle (SQLLDR). SQL\*Loader is used for loading data extracted from SAP to the Oracle staging area.

## Getting the Right Privileges

The SAP Adapter requires privileges to perform set up and execution operations. Please provide your administrators with the list of privileges listed in [Appendix A, "SAP ABAP ERP Required Privileges."](#) These privileges are required for the SAP user that they will provide you to login the SAP System.

## Gathering SAP Connection Information

In order to connect to the SAP ERP system, you must request the following information from your SAP administrators:

- **SAP ERP System IP Address or Hostname:** IP address/ Hostname is the technical name given to the host on which SAP is running.
- **SAP User:** SAP User is the unique user name given to a user for logging on the SAP System.
- **SAP Password:** Case-sensitive password used by the user to log in.
- **SAP Language:** Code of the language used when logging in For example: EN for English, DE for German.
- **SAP Client Number:** The three-digit number assigned to the self-contained unit which is called *Client* in SAP. A Client can be a training, development, testing or production client or represent different divisions in a large company.
- **SAP System Number:** The two-digit number assigned to a SAP instance which is also called Web Application Server or WAS.
- **SAP System ID:** The three-character, unique identifier of a SAP system in a landscape.
- **SAP SNC Connection Properties (optional) SAP Router String (optional):** SAP is enhancing security through SNC and SAP router. It is used when these securities are implemented.
- **SAP Transport Layer Name:** This string uniquely identifies a transport layer in a SAP landscape. It allows ODI to create transport requests for later deployment in SAP. Even though there is a default value here, this transport layer name **must** be provided by your SAP Basis team. Not doing so may result in significant delays during installation.
- **SAP Version:** The version of the SAP system.
- **SAP Character Set:** The character set is only required if your SAP system is not a UNICODE system. For a complete list of character sets, see "Locale Data" in the Oracle Database Globalization Support Guide. For example, EE8ISO8859P2 for Croatian Data. For UNICODE systems, use UTF8.

---

---

**Note:** All the connection data listed above (except SAP SNC Connection Properties and SAP Router String) are mandatory and should be requested from the SAP Administrators. You may consider requesting support during connection setup from your SAP administrators.

---

---

## Validating the Shared Folder Setup

Validating a shared folder setup applies only if you plan to transfer data using a Shared Directory. This section can be skipped, if the FTP transfer is used. The validation of the shared folder setup needs to be performed before any subsequent steps in this guide can be performed. This validation is typically performed by your SAP Basis team.

### Step 1: Validating folder access from the SAP application server

1. Start the SAPGUI.

2. Use the ODI SAP user and password to connect to the SAP system and client.
3. Go to transaction AL11.
4. Select the shared folder.
5. Double-click the shared folder to test the directory declaration.

This should report a successful connection. If not, please contact your SAP basis team. Do not continue until this test passes.

---

---

**Note:** SAP uses the OS user <sid>adm to connect to the directories declared in AL11.

---

---

6. Print screen.

### **Step 2: Validating folder access from the ODI agent machine**

1. Log in to the ODI agent machine using the Windows user ID used for executing the ODI agent.
2. Open shared directory path in Windows Explorer.

This should list the content of the shared folder. Please make sure that you do NOT need to enter any credentials. If this does not work or you had to enter credentials, please contact your windows administrator or the system administrator of the system your shared folder is physically located on. Do not continue until this test passes.

3. Print screen.

## **Validating the FTP Setup**

Validating an FTP setup applies only, if you plan to transfer data using FTP. This section can be skipped, if you use a Shared Directory for the data transfer. The validation of the FTP setup needs to be performed before any subsequent steps in this guide can be performed. This validation is typically performed by your SAP Basis team.

### **Step 1: Validating SAPFTP destination**

1. Start the SAPGUI.
2. Use the ODI SAP user and password to connect to the SAP system and client.
3. Go to transaction SM59.
4. Expand TCP/IP connections.
5. Open the SAPFTP destination.
6. Click **Testconnection**.

This should report a successful connection. If not, please contact your SAP basis team. Do not continue until this test passes.

7. Print screen.

### **Step 2: Testing FTP connection**

1. Go to transaction SE38.
2. View the function module RSFTP002.

3. Hit F8 to run the ABAP program.
4. Enter the FTP userID and password.
5. Enter the FTP server host name or IP address.
6. Enter `cd /` or `cd <ODI target directory>`.
7. In the RFC\_DESTINATION field, enter `SAPFIPA`.
8. Hit F8 to run the test.

This should report a successful connection. The message should be similar to the following:

```
250 CWD successful.
```

If this test is not successful, please contact your SAP basis team. Do not continue until this test passes.

9. Print screen.

## Validating SAP Privileges

This section describes how to test some of the key SAP privileges. Proceed with the subsequent steps in this guide only after successful validation of these tests. This validation is typically performed by your SAP Basis team.

Perform the following steps to validate whether a SAP user has appropriate dev rights and owns a dev license key:

1. Start SAPGUI.
2. Use the ODI SAP user and password to connect to the SAP system and client.
3. Go to transaction SE38.
4. Enter any sample program name like `ZSAP_TEST` in the program name field.
5. Click **Create**.
6. Perform similar tests for the transaction SE37 and SE11.

If a transaction allows the creation of a program without asking for any key or other authorization message, then the SAP user has validated that it has the appropriate dev rights and license key. Otherwise your SAP basis team needs to register the SAP user in `service.sap.com` to get the license key and a Basis person can help him with dev rights.

## Validating SAP Transport Layer Name

As the SAP connector creates SAP objects, such as, for example, function modules, into the SAP development system, these changes need to be transported into QA and production systems once the development is done. The SAP's change and transport system uses the *SAP Transport Layer Name* to identify the route a change has to take. A transport layer is assigned to each development class and thus to all objects in that class. The transport layer determines:

- In which SAP System developments or changes to the repository objects are made
- If objects are transported to other systems within the group when development work has been completed

A consolidation route is created from the development system to the quality assurance system through the transport layer Z<SID>. It then becomes the standard transport layer for customer development and customizing.

A consolidation route is created from the development system to the quality assurance system through the transport layer SAP for the transport of SAP Standard objects.

It is important to specify the correct transport layer name before running the RKM SAP ERP for the first time. Perform the following steps to identify the list of defined transport layers in your SAP landscape:

1. Log on in client 000 in the SAP System serving as the transport domain controller via transaction STMS.
2. Select **Overview > Transport Routes**. The Display Transport Routes dialog is displayed.
3. Select **Goto > Graphical Editor**.
4. To switch the mode, select **Configuration > Display <-> Change**.
5. Position the cursor on the SAP System.
6. Select **Edit > System > Change**. The Change System Attributes dialog is displayed.
7. Select the StandardTransport Layer tab.
8. Change the transport layer of the SAP System.
9. The result is the list of the different transport Layers.

By default, the RKM option SAP\_TRANSPORT\_LAYER\_NAME is set to SAP. Ask your SAP basis admin which transport layer you should use. This transport layer name must be set on the SAP\_TRANSPORT\_LAYER\_NAME RKM option. A wrong or invalid transport layer name will cause serious delays during the installation process.

## Installing Oracle Data Integrator

Before starting with this project, you need to install and configure Oracle Data Integrator. See the *Oracle Fusion Middleware Getting Started with Oracle Data Integrator* and the *Oracle Fusion Middleware Installation Guide for Oracle Data Integrator* for more information.

## Installing and Configuring the Oracle DB Target

This project uses an Oracle database engine as the target and as the staging area for the integration interfaces. You can download for example Oracle Database 11g Express Edition from Oracle Technology Network (OTN). Install and configure this database.

This project targets an Oracle table that can be created using the following script:

```
-- Create demo target schema
CREATE USER ODI_SAP_DEMO IDENTIFIED BY ODI_SAP_DEMO DEFAULT TABLESPACE USERS
TEMPORARY TABLESPACE TEMP;
GRANT CONNECT, RESOURCE TO ODI_SAP_DEMO;

-- Create demo target table
CREATE TABLE ODI_SAP_DEMO.W_GEO_DS (
    LANGUAGE_KEY CHAR(20),
    COUNTRY VARCHAR(50),
    COUNTY VARCHAR(50),
```

```
STATE_PROV VARCHAR(50),
NATIONALITY VARCHAR(50));

ALTER TABLE ODI_SAP_DEMO.W_GEO_DS ADD CONSTRAINT PK_W_GEO_DS PRIMARY KEY
(LANGUAGE_KEY);
```

## Installing and Configuring JCo

The SAP adapter uses Java Connector (JCo) to connect to the SAP system. JCo must be configured before proceeding with the project.

To install and configure JCo:

1. Download a supported JCo version for your configuration from <http://service.sap.com/connectors>. Check the supported JCo version in the Certification Matrix available at Oracle Technology Network. Note that a minimum version of JCo 3.0.2 is required.

---



---

### Notes:

- Choose the SAP JCo package matching your operating system and your system architecture (32/64Bit). E.g. if you are running ODI inside a 32-Bit JVM, you must download the 32-Bit SAP JCo, even if the CPU and OS are 64-Bit. Mixing 32-bit and 64-bit architecture is not possible due to native libraries required by SAP JCo and will result in connection failure.
  - `odi.conf` contains the JDK path used for ODI Studio.
- 
- 

2. Unzip the appropriate distribution package into an arbitrary directory `<sapjco-install-path>`.
3. Follow the installation instructions in `<sapjco-install-path>/javadoc/installation.html` for the respective platform.
4. Copy `sapjco3.jar` and `sapjco3.dll`:  
 For ODI 10g: Copy `sapjco3.jar` and `sapjco3.dll` (or respective binary like `libsapjco3.so`) into the `oracledi/drivers` directory.  
 For ODI 11g: Copy `sapjco3.jar` and `sapjco3.dll` (or respective binary like `libsapjco3.so`) into the `<ODI_HOME>/odi_misc` directory (ODI Studio) and into the `<ODI_HOME>/oracledi/agent/drivers` directory (Standalone Agent).
5. Restart ODI Studio and any agents.
6. Check the JCo installation. This will be also checked later in this guide.

---



---

**Caution:** Changing the JCo library installed in the `<windows-dir>\system32` directory of a machine running other SAP tools or components may cause issues with these components. Please check with your machine administrator before performing this change.

---



---

---

---

**Note:** Note: The SAP JCo software requires the `librfc32/librfc64` library, but does not include it. It is included e.g. with the SAP GUI software. In case this library is not yet installed, it has to be downloaded and installed following the steps outlined in [Appendix B, "Installing the librfc32/librfc64 Library."](#)

---

---

## Setting Up an FTP Server

The SAP adapter extracts SAP data and uploads it to an FTP server. Configure an FTP server or use an existing FTP server. You must create a user in this FTP server and a directory into which this user will be able to upload files.

## Configuring Oracle Data Integrator

Perform the following steps to configure Oracle Data Integrator:

1. [Set Up the Topology](#)
2. [Add the Open Tool](#)

### Set Up the Topology

Perform the following operations after installing or upgrading your Oracle Data Integrator version:

1. Connect to Topology Manager.
2. If this SAP ABAP technology does not exist in your Master Repository, import the SAP ABAP technology in Synonym INSERT\_UPDATE mode from the `/impexp` folder.
3. Perform an upgrade of the Master Repository. Refer to the *Oracle Fusion Middleware Installation Guide for Oracle Data Integrator* for more information on the Master Repository upgrade process.
4. In Topology Manager, open the JavaBeanShell technology and check on the Language tab that the JYTHON language is listed. If not, add it.
5. Create a File data server pointing to an existing FTP server into which the extraction file will be pushed from SAP and picked up for SQL\*Loader. Set the parameters for this data server as follows:
  - **Host (Data Server):** FTP server IP host name or IP address.
  - **User:** Username to log into FTP server.
  - **Password:** Password for the user.
6. For use with "LKM SAP ERP to SQL" these additional parameters must be configured:
  - JDBC driver class: `com.sunopsis.jdbc.driver.file.FileDriver`
  - JDBC URL: `jdbc:snps:dbfile?ENCODING=UTF8`

The above URL is for SAP UNICODE systems. For non-UNICODE systems, please see details on ENCODING parameter in section 3.3.1, "Creating a File Data Server" of *Oracle Fusion Middleware Connectivity and Knowledge Modules Guide for Oracle Data Integrator*. The encoding chosen on this URL must match the code page used by the SAP Application Server.

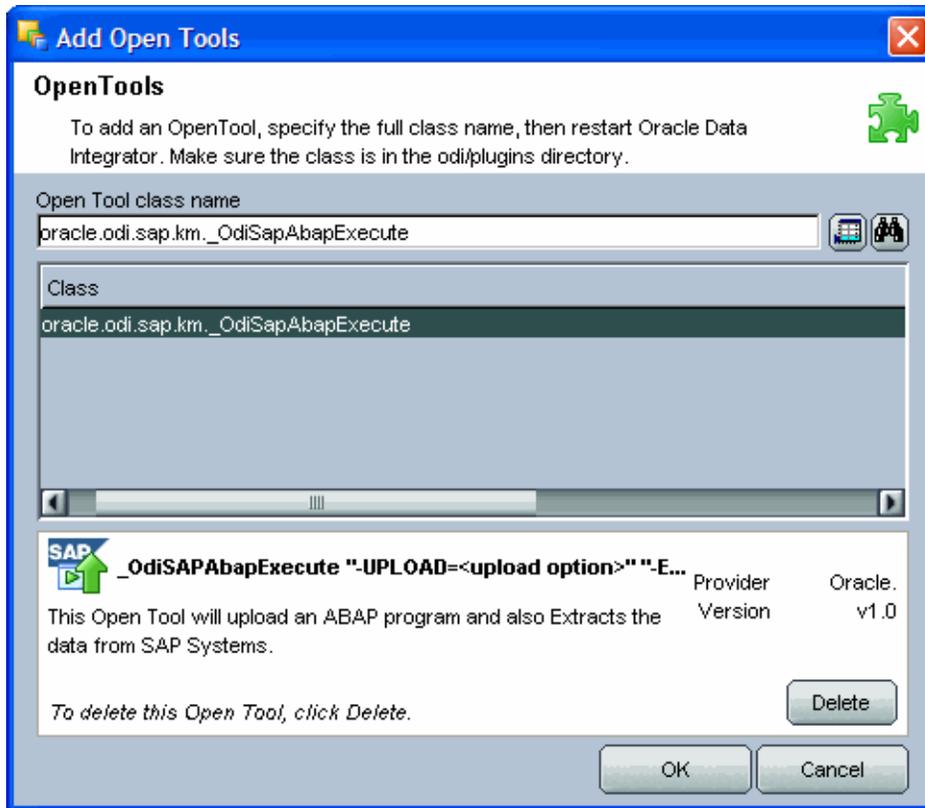
7. In this File data server create a Physical Schema representing the folder in the FTP host where the extraction file will be pushed. Specify the Data and Work Schemas as follows:
  - **Data Schema:** Path on the FTP server to upload or download extraction files from the remote location. This path is used when uploading extraction files from the SAP ERP system into the FTP server. It is also used by a remote agent to download the extraction files. Note that this path must use slashes and must end with a slash character.
  - **Work Schema:** Local path on the FTP server's machine. This path is used by an agent installed on this machine to access the extraction files without passing via the FTP server. This access method is used if the FTP\_TRANSFER\_METHOD parameter of the LKM SAP ERP to Oracle (SQLLDR)/ LKM SAP ERP to SQL is set to NONE. As Work Schema is an OS file name, slashes/backslashes should be used according to OS.  
  
Path names need to end on slash/ backslash. Path names given on Data and Work schemas are not necessarily the same: the FTP service may provide access to a FTP directory named /sapfiles while the files can be stored locally in c:\inetpub\ftproot\sapfiles.

Refer to Section 8.7.1, "File Transfer Considerations" of *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator*.
8. If the corresponding Logical Schema called File Server for SAP ABAP does not exist yet, create it. This Logical Schema name is fixed and must be mapped to the Physical Schema created in the previous step.

## Add the Open Tool

The complete process of installing and adding an Open Tool to ODI is described in Appendix A, "Oracle Data Integrator Tools Reference" of *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*. This section details only the SAP ABAP specific steps.

1. Connect to Designer.
2. Select **File > Add/Remove Open Tools...**
3. In the Add/remove Open Tools window, enter the following name in the Open Tool class name field: `oracle.odi.sap.km._OdiSapAbapExecute`
4. Click **Add Open Tool**.
5. The Open Tool appears as shown below.



Add Open Tools window. An Open Tool class name is shown in a field with lookup and search buttons. Below is the name of the Class, followed by a section describing the Open Tool. This section contains a Delete button for deleting this Open Tool. At the bottom are buttons marked OK and Cancel.

\*\*\*\*\*

6. Click OK.

---

---

## Configuring the Topology

This chapter describes how to configure the topology by using a practical example.

This project will integrate the following source and target data:

Source tables *T005T* and *T005U* are the tables from SAP containing geographical information.

Target *W\_GEO\_DS* is a dimension table in an Oracle database that needs to be loaded from the SAP Sources. This table contains geographical information such as Continent name, Country name, Region, State, County, City and Zip code.

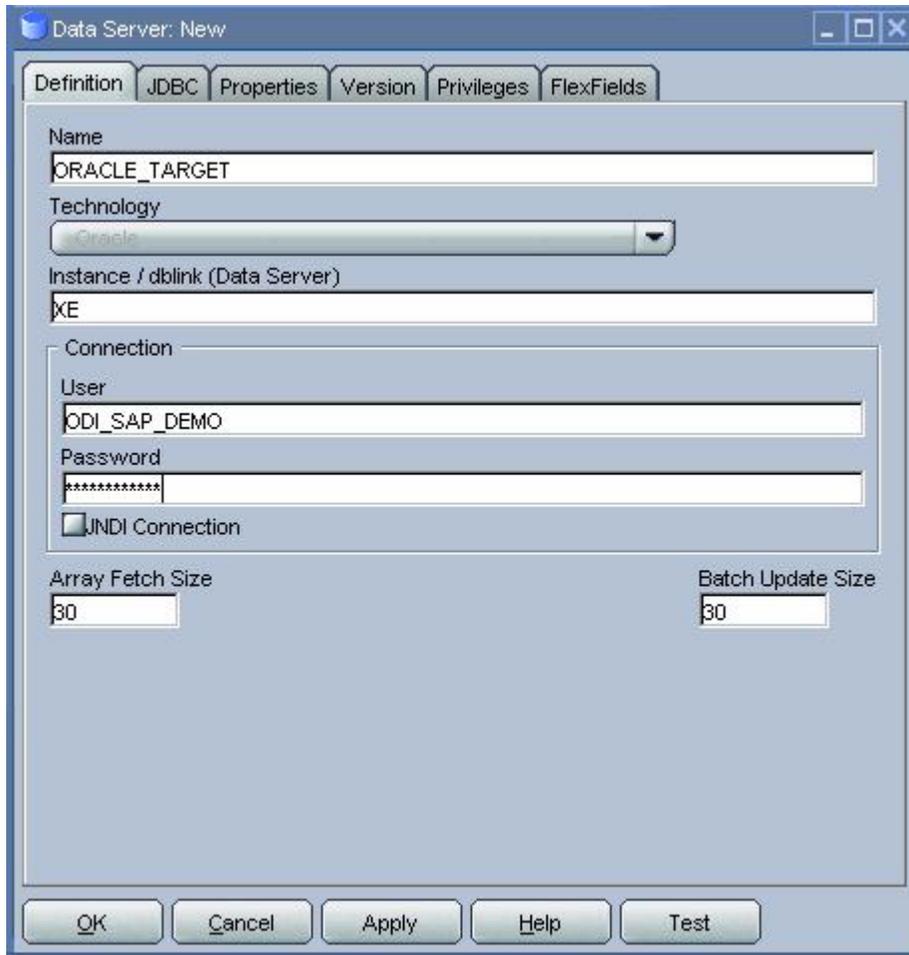
This chapter includes the following topics:

- [Configuring the Target Data Server](#)
- [Configuring the SAP Source Server](#)

### Configuring the Target Data Server

This section describes how to setup an Oracle Data Server. Please use respective settings when configuring a non-Oracle Target Data Server for use with LKM SAP ERP to SQL.

1. Connect to Topology Manager.
2. In the Physical Architecture tree view, expand the Technologies node and select the Oracle technology.
3. Right-click and select **Insert Data Server**.
4. Enter the data server Definition as shown below:



The Data Server: New window is shown. The Definition tab is selected. At the top are Name, Technology, and Instance/dblink (Data Server) fields. A box labeled Connection contains User and Password fields and a check box labeled JNDI Connection. Below are Array Fetch Size and Batch Update Size fields. At the bottom of the window are the following buttons: OK, Cancel, Apply, Help, and Test.

\*\*\*\*\*

5. Select the JDBC tab, and enter the JDBC connection information to your data server as shown below:

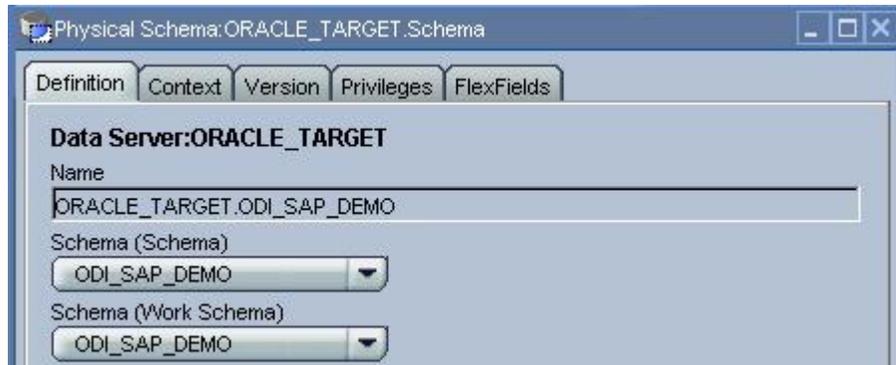


The JDBC tab of the Data Server: ORACLE\_TARGET window is shown. The JDBC Driver field is set to "oracle.jdbc.driver.OracleDriver" and the JDBC Url field is set to "jdbc:oracle:thin:@localhost:1521:XE".

\*\*\*\*\*

Make sure to change the URL to match your Oracle instance configuration.

6. Click **Test** to test the connection to this data server.
7. Click **OK** to save your data server.
8. In the Physical Schema windows that appears, select the ODI\_SAP\_DEMO for both Schema and Work Schema, as shown below:

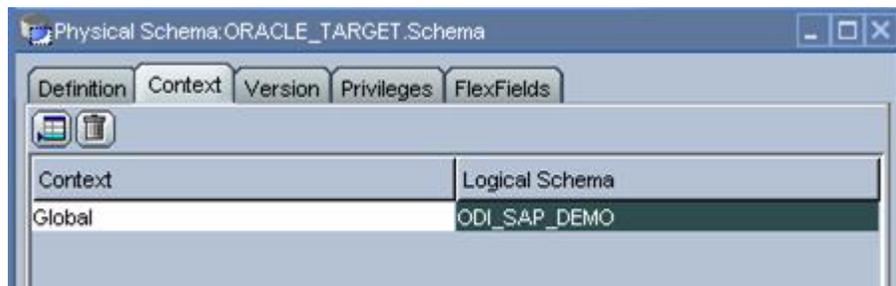


The Physical Schema:ORACLE\_TARGET.Schema window is shown, with the Definition tab selected. The title displayed is "Data Server:ORACLE\_TARGET". The Name field contains the value "ORACLE\_TARGET.ODI\_SAP\_DEMO". The Schema (Schema) field contains the value "ODI\_SAP\_DEMO". The Schema (Work Schema) field contains the value "ODI\_SAP\_DEMO".

\*\*\*\*\*

If you already have a Work Schema for ODI defined for this data server, you can use it instead of the ODI\_SAP\_DEMO Schema.

9. Go to the Context tab.
10. Click **Add** and enter in the Logical Schema field the value ODI\_SAP\_DEMO.



The Context tab of the Physical Schema:ORACLE\_TARGET.Schema window is shown. A table with columns labeled "Context" and "Logical Schema" has a single row: the Context is "Global" and the Logical Schema is "ODI\_SAP\_DEMO".

\*\*\*\*\*

11. Click **OK** to save.

## Configuring the SAP Source Server

The following topics describe how to configure the SAP source server:

- [Configuring the Data Server](#)
- [Configuring the Logical Schema](#)

### Configuring the Data Server

Except for the data server name, all the parameters that you provide while defining the SAP Data server should be provided by the SAP Administrators. See ["Getting the Right Privileges"](#) on page 2-2 for more information.

The SAP Adapter requires privileges to perform set up and execution operations. Please provide your administrators with the list of privileges listed in [Appendix A, "SAP ABAP ERP Required Privileges."](#) These privileges are required for the SAP user that they will provide you to login the SAP System.

See ["Gathering SAP Connection Information"](#) on page 2-3 for more information about these parameters.

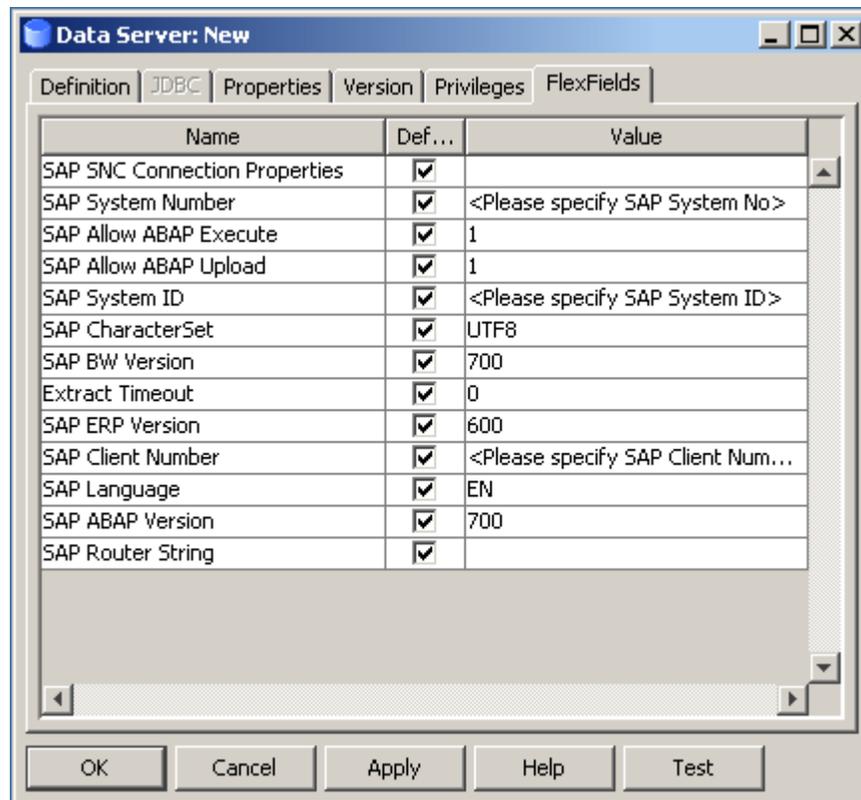
1. In the Physical Architecture tree view, expand the Technologies node, select the SAP ABAP technology.
2. Right-click and select **Insert Data Server**.
3. Enter the data server definition. Set the parameters for this data server as follows:
  - **Name:** SAP\_ERP. The name of the data server as it will appear in ODI.
  - **Host (Data Server):** SAP ERP System IP Address or Hostname.
  - **User:** SAP User, as provided by the SAP Administrator.
  - **Password:** This user's SAP Password. This password is case-sensitive.



The Data Server: New window is shown, with the Definition tab selected. The window displays fields for Name, Technology, Host (Data Server), and (in a section labeled "Connection") User, Password, and a check box labeled JNDI Connection.

\*\*\*\*\*

4. Set the Flexfield values for this data server in the Flexfields tab:



The Data Server: New window is shown, with the FlexFields tab selected. A table is displayed with three columns: Name, Default, and Value.

\*\*\*\*\*

- **SAP Language:** Code of the language used when logging in. For example EN for English, DE for German.
- **SAP Client Number:** The three-digit number assigned to the self-contained unit which is called *Client* in SAP. A Client can be a training, development, testing or production client or represent different divisions in a large company.
- **SAP System Number:** The two-digit number assigned to a SAP instance which is also called Web Application Server or WAS.
- **SAP System ID:** The three-character, unique identifier of a SAP system in a landscape.
- **SAP SNC Connection Properties:** SNC Connection Properties. This parameter is optional and can be left empty.
- **SAP Router String:** Router String. This parameter is optional and can be left empty.
- **SAP Character Set:** The character set is only required if your SAP system is not a UNICODE system. For a complete list of character sets, see "Locale Data" in the Oracle Database Globalization Support Guide. For example, EE8ISO8859P2 for Croatian Data. For UNICODE systems, use UTF8.

---

---

**Note:** This FlexField is only used with "LKM SAP BW to Oracle (SQLDR)". For use with "LKM SAP BW to SQL" please define the character set using the ENCODING parameter in the file driver URL, as described in "Set Up the Topology" on page 2-8 for non-UNICODE SAP systems.

---

---

- **SAP ABAP Version:** Enter the SAP ABAP version as follows:
    - For SAP 4.6C enter 46C
    - For SAP 4.7 enter 620
    - For SAP ECC 5.0 enter 640
    - For SAP ECC 6.0 enter 700
  - **SAP BW Version & SAP ERP Version:** unused in SAP ERP Connector.
5. Click **OK**.

---

---

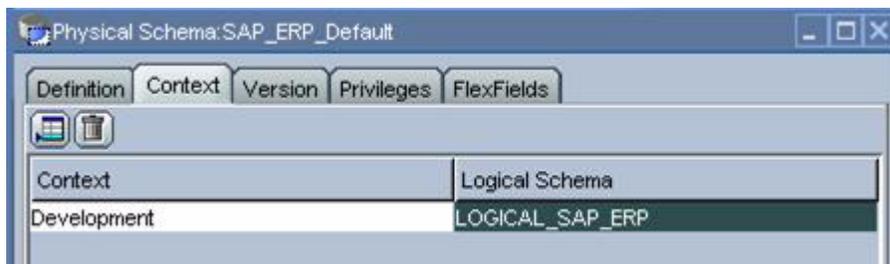
**Note:** The **Test** button for validating SAP Connection definition is not supported for this connection.

---

---

## Configuring the Logical Schema

1. In the Physical Schema window, do not edit the Definition tab.
2. Select the Context tab, click **Add** and enter the Logical Schema name LOGICAL\_SAP\_ERP.



The Context tab of the Physical Schema:SAP\_ERP\_Default window is shown. A table is displayed with two columns: Context and Logical Schema. A single row has the value "Development" for Context and the value "LOGICAL\_SAP\_ERP" for Logical Schema.

\*\*\*\*\*

3. Click **OK**.

---

---

## Creating a New Project

This chapter describes how to create a new Project for SAP.

In order to work with the data servers you have created, you must create a project and import the appropriate knowledge modules for reverse-engineering the data structures and integrating the data.

### Creating a New Project

To create a new Project for SAP:

1. Connect to Designer.
2. In the Projects tree view click **Insert Project**.
3. In the Definition tab, enter `SAP Demo` for the Project name.
4. Click **OK** to save your project.
5. Expand the `SAP Demo Project`, select the `First Folder` node.
6. Rename this folder to `SDE_SAP_GeoDimension`.
7. Right-click the `SAP Demo Project` in the Projects tree view.
8. Right-click and select **Import Knowledge Modules**.
9. Select in the File import directory the directory containing your KMs. By default, they are located in the `../impexp` directory.
10. Select the following KMs:
  - IKM Oracle Incremental Update
  - LKM SAP ERP to Oracle (SQLLDR)
  - LKM SAP ERP to SQL
  - RKM SAP ERP
  - RKM SAP ERP Connection Test
11. Click **OK** to perform the import.



---

## Reverse-Engineering Data Models

This chapter describes how to reverse-engineer the Oracle target and SAP source data models.

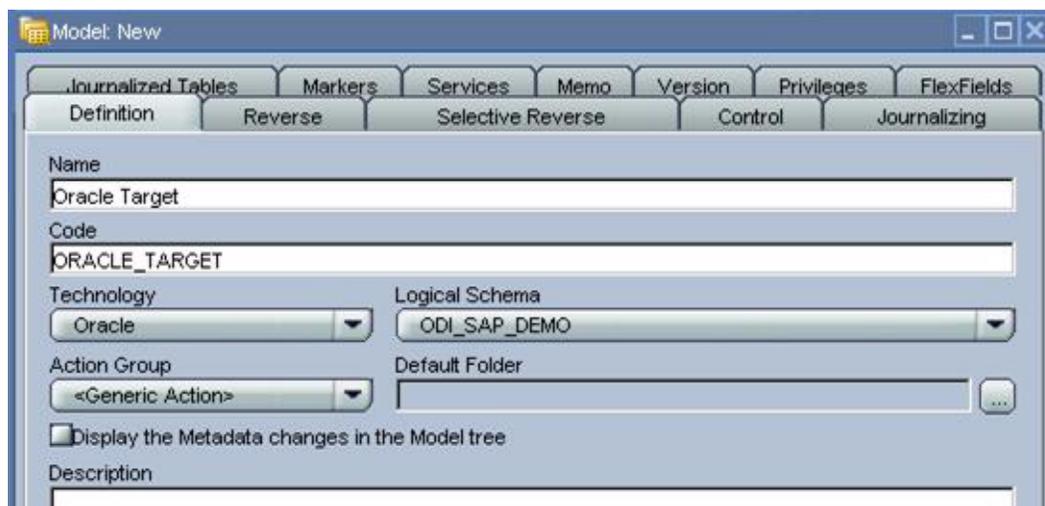
It contains the following topics:

- [Reverse-Engineering the Oracle Target](#)
- [Reverse-Engineering the SAP Source Datastores](#)

### Reverse-Engineering the Oracle Target

To reverse-engineer the Oracle data model:

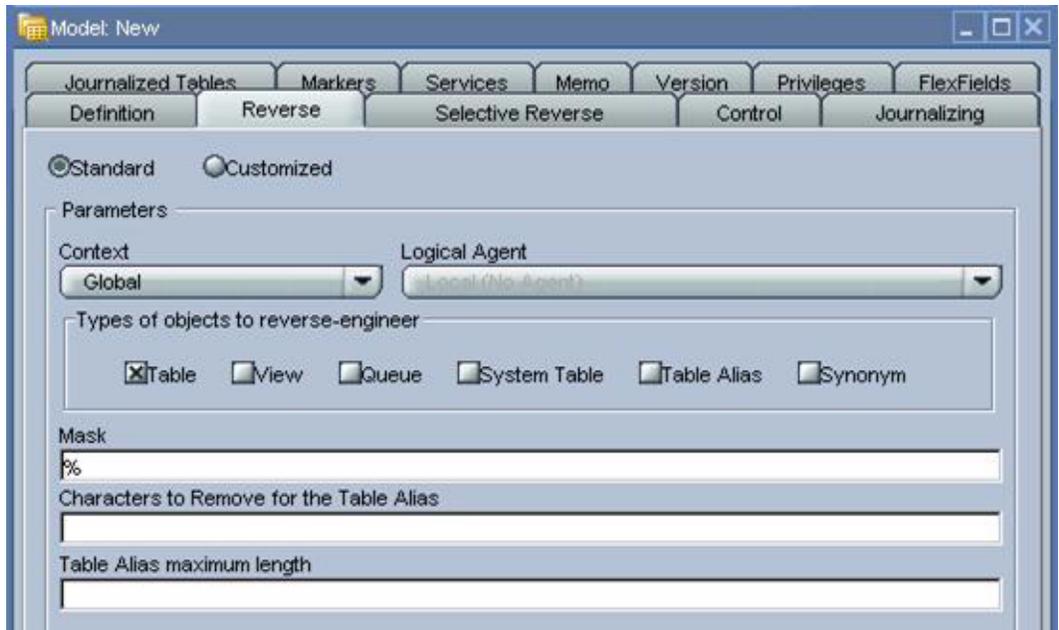
1. Connect to Designer.
2. In the Models tree view, click **Insert Model**.
3. In the Definition tab, enter the model parameters as shown below:



The Model: New window is shown, with the Definition tab selected. The Name field has the value "Oracle Target". The Code field has the value "ORACLE\_TARGET". The Technology field has the value "Oracle". The Logical Schema field has the value "ODI\_SAP\_DEMO". The Action Group field has the value "<Generic Action>". There is no value entered for the Default Folder field. The Display the Metadata Changes in the Mode tree check box is not checked.

\*\*\*\*\*

4. In the Reverse tab, select the **Global** context.



The Model: New window is shown, with the Reverse tab selected. The Standard/Customized radio button is set to Standard. In the Parameters box, the Context field is set to "Global". The value for Logical Agent is greyed out. In the Types of objects to reverse-engineer box, the Table check box is checked; all other boxes are empty. The Mask field has the value "%" (percent sign). The remaining fields are empty.

\*\*\*\*\*

5. Click **Reverse**, then **OK** to save your model.
6. The WS\_GEO\_DS datastore appears in your Oracle model, as shown below.



A tree diagram is displayed. The root is "Oracle Target", which has been expanded. The following un-expanded branches below "Oracle Target" are shown: "Used in", "Diagrams", "Hierarchy", "W\_GEO\_DS", and "Hidden Databases".

\*\*\*\*\*

## Reverse-Engineering the SAP Source Databases

To reverse-engineer the SAP Source databases you need to perform the following tasks:

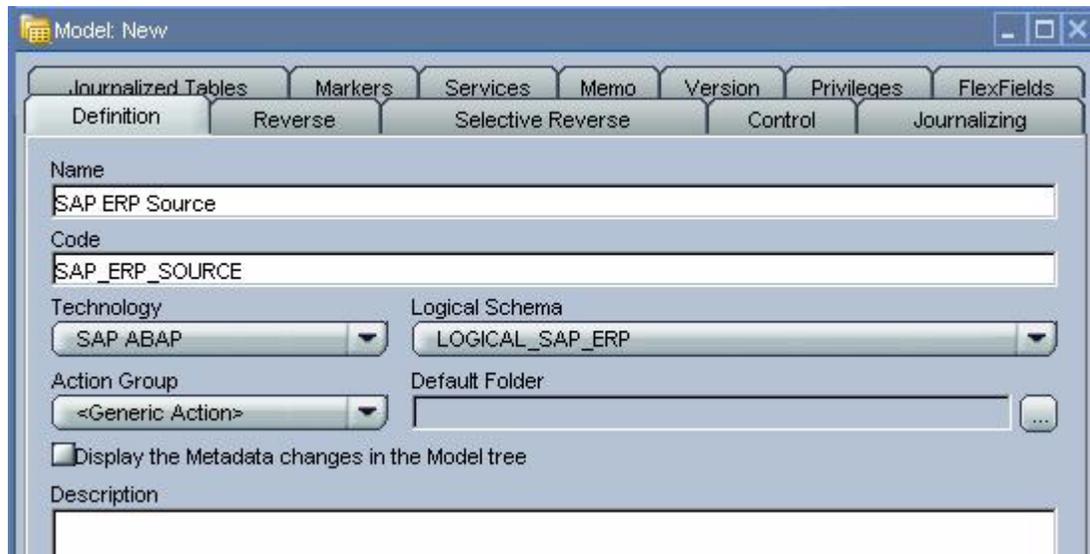
1. [Creating the Data Model](#)
2. [Validating the SAP Connection](#)
3. [Starting the Reverse-Engineering Process](#)

This section also includes the following topics:

- [Using the SAP Metadata Browser](#)
- [Reverse-Engineering Without GUI](#)

## Creating the Data Model

1. In the Models tree view, click **Insert Model**.
2. In the Definition tab, enter the model parameters as shown below. You must select the SAP ABAP technology and the Logical Schema previously created (in our example it is the LOGICAL\_SAP\_ERP Logical Schema).



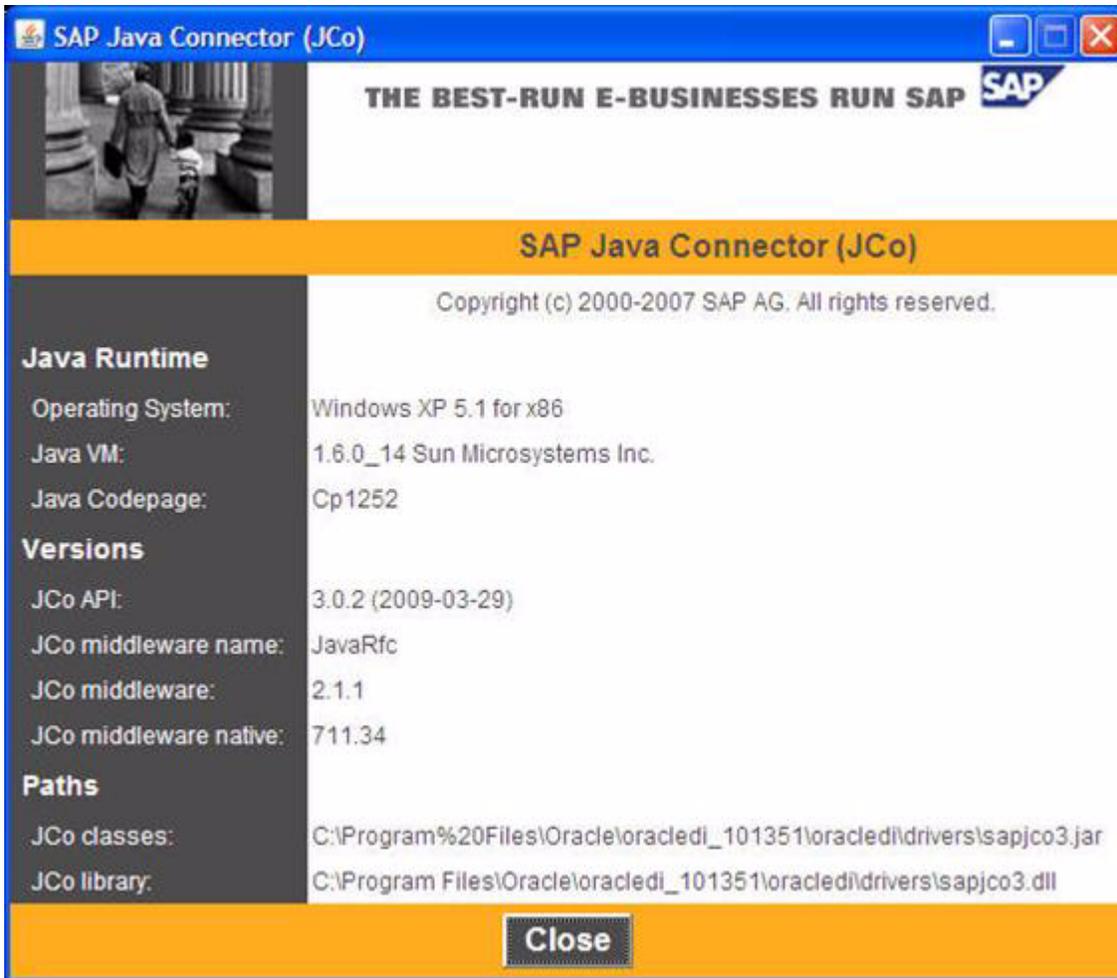
The Model: New window is shown, with the Definition tab selected. The Name field has the value "SAP ERP Source". The Code field has the value "SAP\_ERP\_SOURCE". The Technology field has the value "SAP ABAP". The Logical Schema field has the value "LOGICAL\_SAP\_ERP". The Action Group field has the value "<Generic Action>". The Default Folder field has no value. The Display the Metadata changes in the Model tree check box is unchecked.

\*\*\*\*\*

3. In the Reverse tab:
  1. Select the Global context.
  2. Select Customized option.
  3. Select the RKM SAP ERP Connection Test you have imported in the SAP ERP Demo project.

## Validating the SAP Connection

1. In the Models tree view, open the SAP ERP Source Model.
2. Click **Reverse** and confirm the start of the reverse-engineering process. This process runs two tests:
  - Establish a test connection to the SAP system and
  - Validate the proper setup of SAP JCo by displaying the About dialog.
3. In a few seconds, you should see the SAP JCo-About dialog as shown here:



The SAP Java Connector (JCo) window is shown. Values are shown for Java Runtime Operating System, Java VM, and Java Codepage: Versions for JCo API, JCo middleware name, JCo middleware, and JCo middleware native: and Paths for JCo classes and JCo library. At the bottom of the window is a Close button.

\*\*\*\*\*

Please verify the JCo version number (see "System Requirements and Certification" on page 2-1 for more details) and verify that a path for the JCo Library is given (.dll for Windows). If this is not the case or you see any error message in this about dialog, please review the SAP JCo installation, as described in your SAP JCo package, and run this test again.

---

**Note:** Closing the JCo-About-Dialog will also close ODI.

---

4. Once you have validated SAP JCo installation, open ODI Operator.
5. In the Operator, expand the Reverse-Engineering session down to the task level.
6. Verify whether the task "Test SAP Connection" has been executed successfully. If not, view the task details to identify the connection problem. The output of this task *must* contain Successfully connected to SAP System. If not, the connection test has failed and the connection problem must be analyzed *before* you continue.

---

**Do not continue until you have successfully validated the SAP connection!**

---

**Note:** If you want to validate the SAP connectivity independent of ODI, please see [Appendix C, "SAP Stand-Alone Connection Test."](#)

---

## Starting the Reverse-Engineering Process

To start the reverse-engineering process of the SAP ERP datastores:

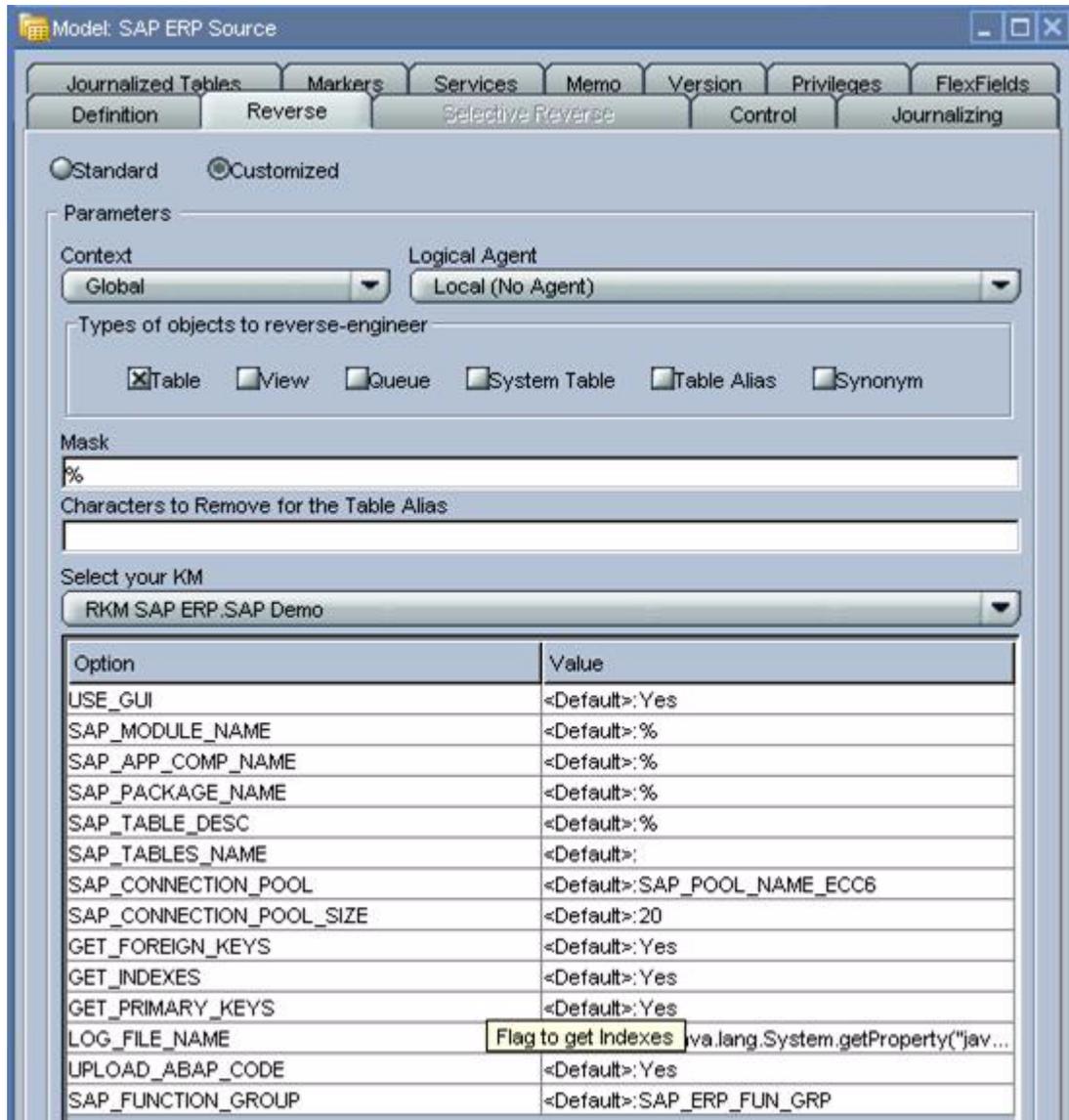
1. In the Models tree view, open the SAP ERP Source Model.
2. In the Reverse tab:
  - a. Select the Global context.
  - b. Select the Customized option.
  - c. Select the RKM SAP ERP you have imported in the SAP ERP Demo project.
  - d. Set the USE\_GUI KM option to Yes.
  - e. For first time RKM use only: Set SAP\_TRANSPORT\_LAYER\_NAME to the name of the transport layer your SAP basis admin has provided you with.

---

**Caution:** Pay particular attention to provide the correct value for the SAP\_TRANSPORT\_LAYER\_NAME option. If an invalid value is used during the first time installation, a significant delay in the connector installation may be caused.

---

- f. For first time RKM use only: If you are using a custom value for KM option SAP\_FUNCTION\_GROUP\_NAME, this function group must *not* exist prior the first execution of the RKM (ABAP\_UPLOAD=true).
- g. For first time RKM use only: Set UPLOAD\_ABAP\_CODE to Yes. **After the first execution this value must be set back to No.**
- h. Set other parameters according to your SAP configuration. See the *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator* for more information on the RKM options.



The Model: SAP ERP Source window is shown, with the Reverse tab selected. In the lower half of the window, the Select your KM field has the value "RKM SAP ERP.SAP Demo". A table with the column headings "Option" and "Value" has numerous rows with option/value pairs.

\*\*\*\*\*

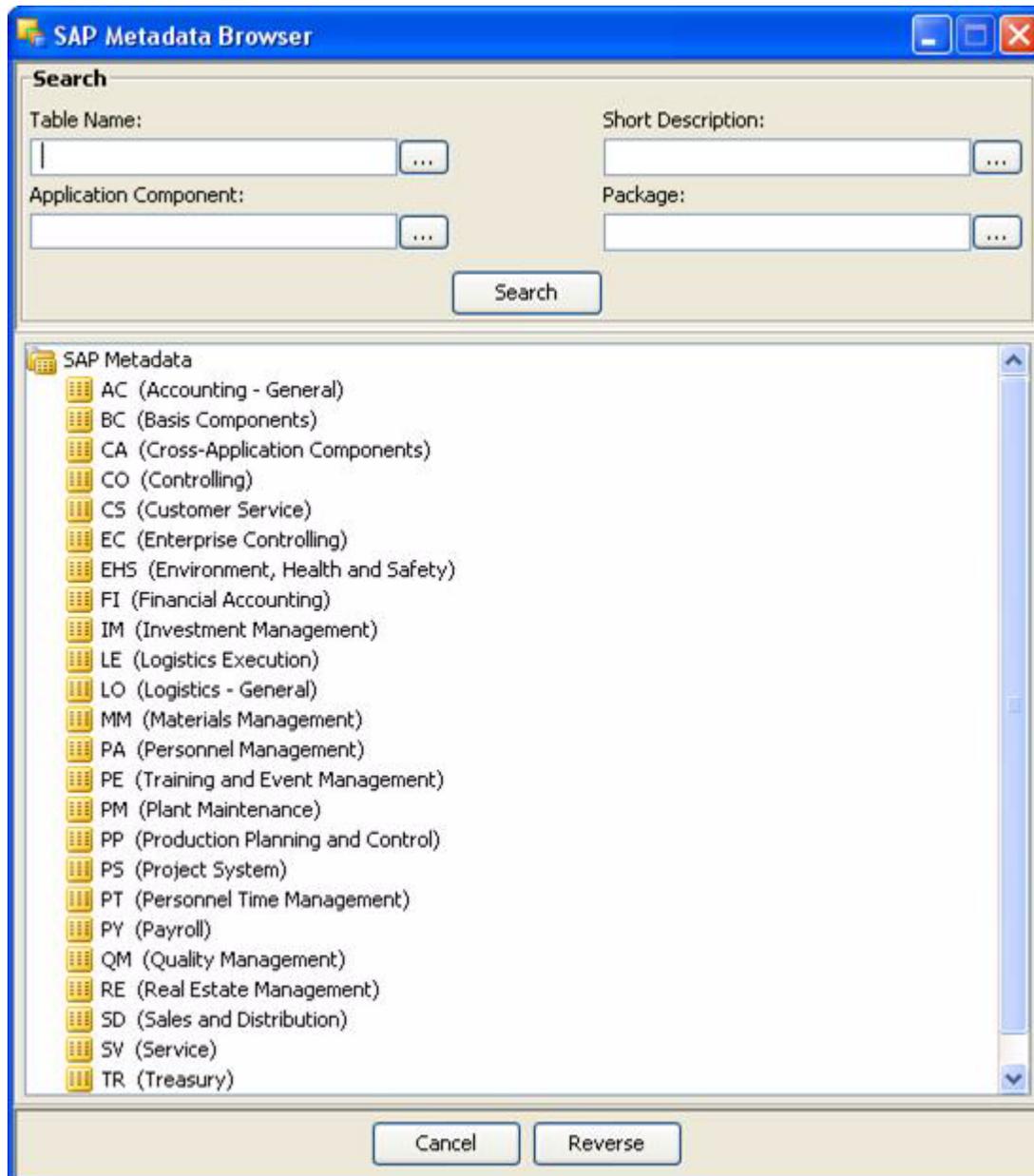
3. Click **Apply** to save your changes.
4. Click **Reverse** to start the reverse-engineering process.
5. Click **OK**.
6. The Sessions Started Dialog is displayed.
7. Click **OK**.
8. Set UPLOAD\_ABAP\_CODE back to No.
9. For first time RKM use only: Validate in Operator that the session is now in status running. If session has failed, please validate settings. Do not move on until all installation steps have been completed successfully.

## Using the SAP Metadata Browser

As you have set the USE\_GUI option to Yes, the RKM displays in the **SAP Metadata Browser** the SAP objects organized in the tree view as shown below.

**Note:** In case that the below dialog does not show up, see ODI Operator for details and read the log file <System Temp Dir>/sap\_rkm\_erp\_<ODI Session Number>.log.

Refer to Section 8.7.4, "Log Files," in *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator* for more details.



The SAP Metadata Browser window is shown. The upper portion of the window has a Search function with fields for Table Name, Short Description, Application Component, and Package. Each field has a "..." button. The lower portion of the

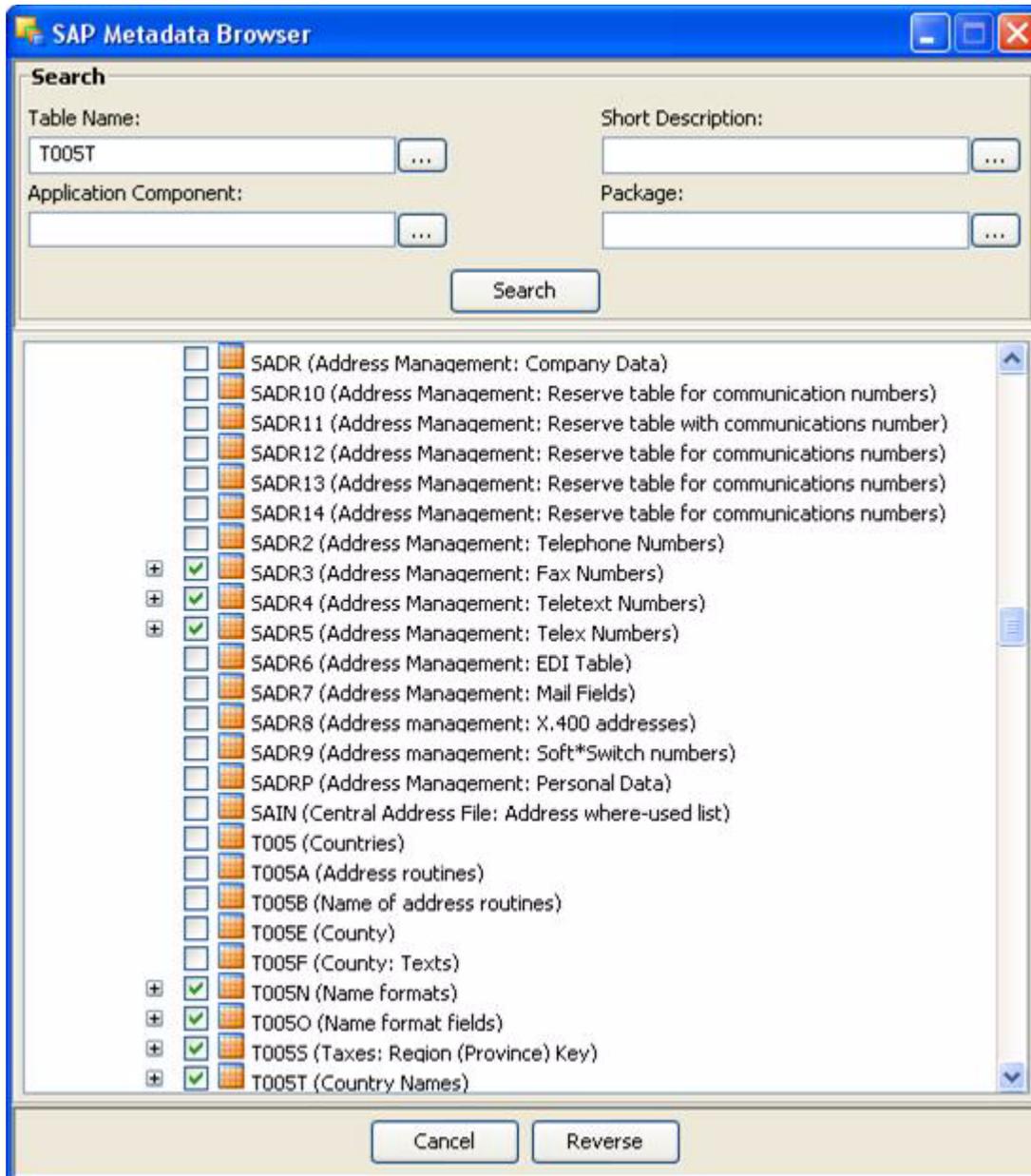
window has an expanded tree view. A single object, "SAP Metadata", is expanded with many child objects with names like "AC (Accounting - General)" and "BC (Basis Components)". At the bottom of the window are buttons labeled Cancel and Reverse.

\*\*\*\*\*

### The Tree View

The SAP Modules are organized into a tree view. Only the SAP Modules available to the current user are displayed. By expanding the tree view, you can see SAP tables, and select those that you want to reverse-engineer.

Note that only the tables selected in the tree view are reverse-engineered.



The SAP Metadata Browser window is shown. In the Search box at the top, the Table Name field has the value "T005T". In the expanded tree view, many objects are listed,

each with a check box. A few objects have checked check boxes; these objects also have "plus" icons indicating they are expandable branches.

\*\*\*\*\*

### The Search Panel

The **Search** option group allows you search specific objects in the browser, based on:

- Table Name
- Short Description
- Application Component
- Package

For the example project, the T005T and T005U country tables are used:

1. Enter T005T in the Table Name field and click **Search** to find the first table to reverse-engineer. Select this table.
2. Enter T005U in the Table Name field and click **Search** to find the second table to reverse-engineer. Select this table.
3. Click **Reverse**.
4. Oracle Data Integrator starts the reverse-engineering process on the selected datastores. The reverse-engineered datastores appear under the Model.

## Reverse-Engineering Without GUI

If you want to perform a reverse-engineering without using the SAP Metadata Browser, you can enter the following parameters in the SAP ERP KM options:

- USE\_GUI: No
- SAP\_TABLES\_NAME : T005T, T005U

With this configuration, the tables will be reverse-engineered without displaying the SAP Metadata Browser.

By using the appropriate KM options, you can also select a filter for the reversed tables based on for example a SAP Module or a Package name. You can also specify a table name mask in the **Mask** field of the Reverse tab to filter table names.

Refer to the *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator* for more information on the RKM options.

---

**Note:** It is not possible to view the data in the SAP ERP tables from Oracle Data Integrator.

---

### What happens when you reverse-engineer SAP ERP tables?

First, Oracle Data Integrator uploads in the SAP system ABAP code allowing for retrieving the SAP metadata. This RFC is uploaded only if the UPLOAD\_ABAP\_CODE option is selected (this option is typically activated only once). The ABAP programs are generated in the SAP function group given in the SAP\_FUNCTION\_GROUP RKM option.

Once this code is set up on SAP, Oracle Data Integrator is able to retrieve the SAP metadata and to display it in the SAP Metadata Browser. Selected metadata is then

reverse-engineered from SAP into the Oracle Data Integrator repository. The repository only contains the description of the SAP metadata, and no data at all.

In addition to the table definitions organized and enriched with user-friendly information, the keys and indexes definitions are also retrieved. The model appearing in Oracle Data Integrator shows all the information required for performing integration tasks on the SAP ERP Model.

---

---

## Creating and Running the Integration Interface

This chapter describes how to create and run the integration interface.

It contains the following topics:

- [Creating the Integration Interface](#)
- [Running the Integration Interface](#)

### Creating the Integration Interface

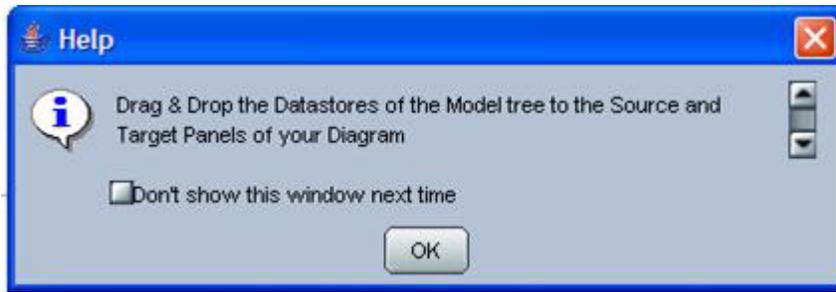
Now that the source and target data models are created, it is possible to create an interface to integrate data from the T005T and T005U SAP tables to the WS\_GEO\_DS Oracle table.

To create the integration interface, perform the following steps:

1. [Create the Interface](#)
2. [Define the Source and Target Datastores](#)
3. [Define Joins between Sources](#)
4. [Create the Mappings](#)
5. [Define the Interface Flow](#)

#### Create the Interface

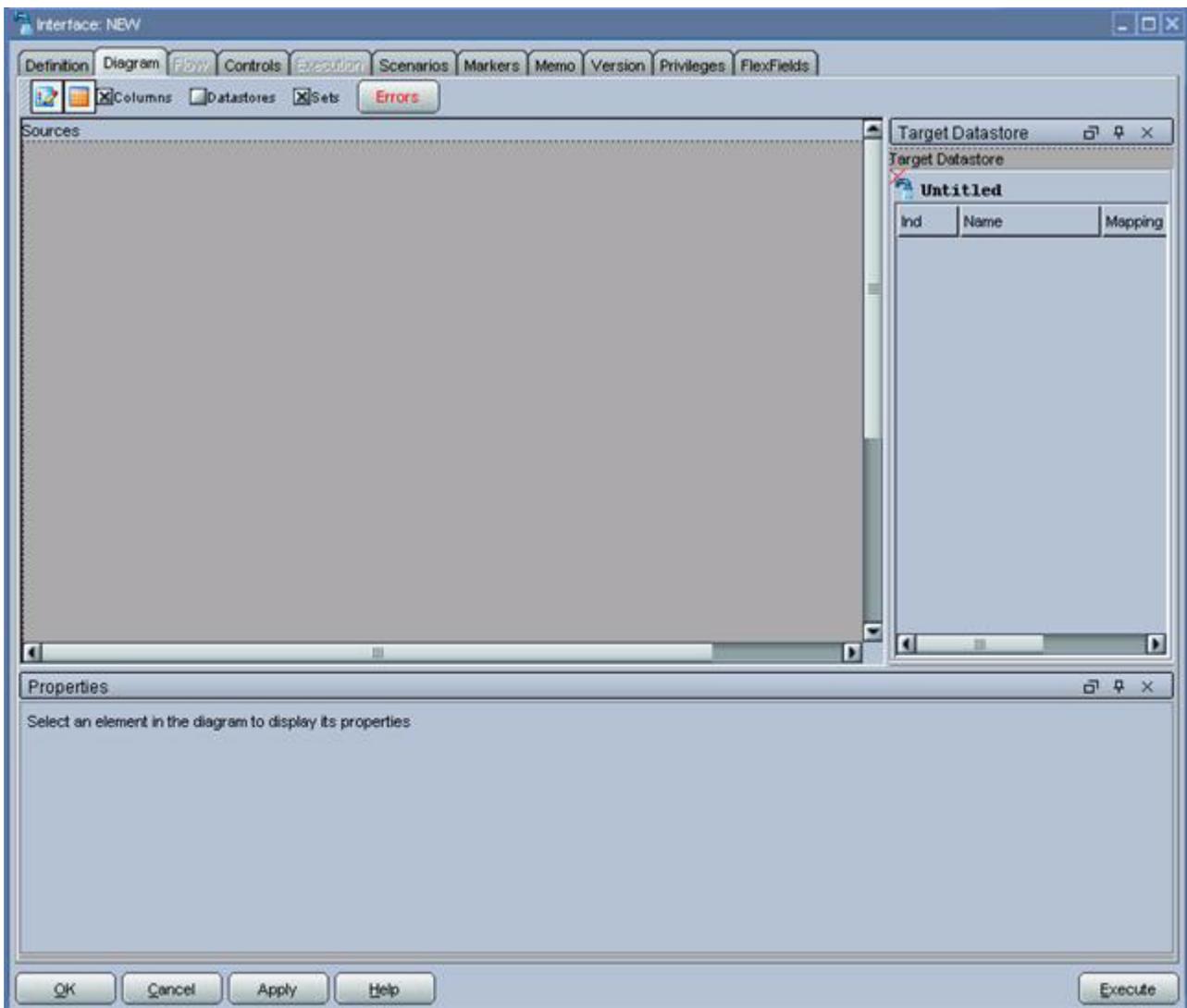
1. Open Designer.
2. In the Projects tree view, expand the SAP Demo Project.
3. Expand the SDE\_SAP\_GeoDimension folder.
4. Select the Interfaces node.
5. Right-click and select **Insert Interface**.
6. In the Interface Definition tab, enter the interface name: SDE\_SAP\_GeoDimension.W\_GEO\_DS.
7. Select the Diagram tab.
8. Click **OK** if the following information window appears.



A Help pop-up dialog box is displayed. It reads "Drag & Drop the Datastores of the Model tree to the Source and Target Panels of your Diagram". It has a check box labeled "Don't show this window next time". At the bottom is an OK button.

\*\*\*\*\*

The interface diagram looks as follows:



The Interface: NEW window is shown, with the Diagram tab selected. A window is shown for displaying a graphical representation of your Sources. On the right, the Target Datastore area shows objects (tables and rows) in your target datastore. The

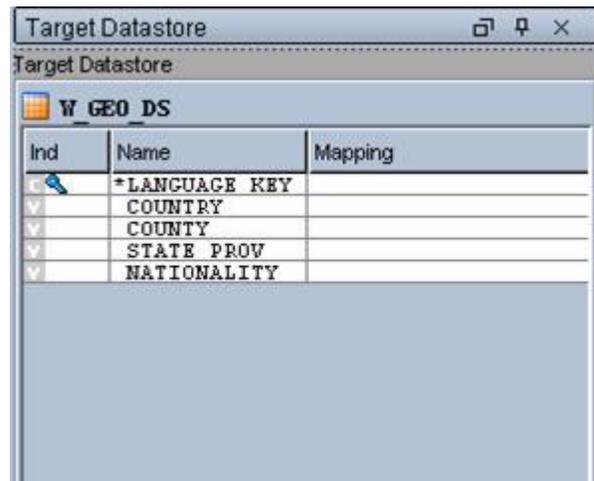
lower area of the window is a Properties section which displays the properties of any selected objects. Along the bottom are buttons labeled OK, Cancel, Apply, Help, and Execute.

\*\*\*\*\*

## Define the Source and Target Datastores

To define the source and target datastores:

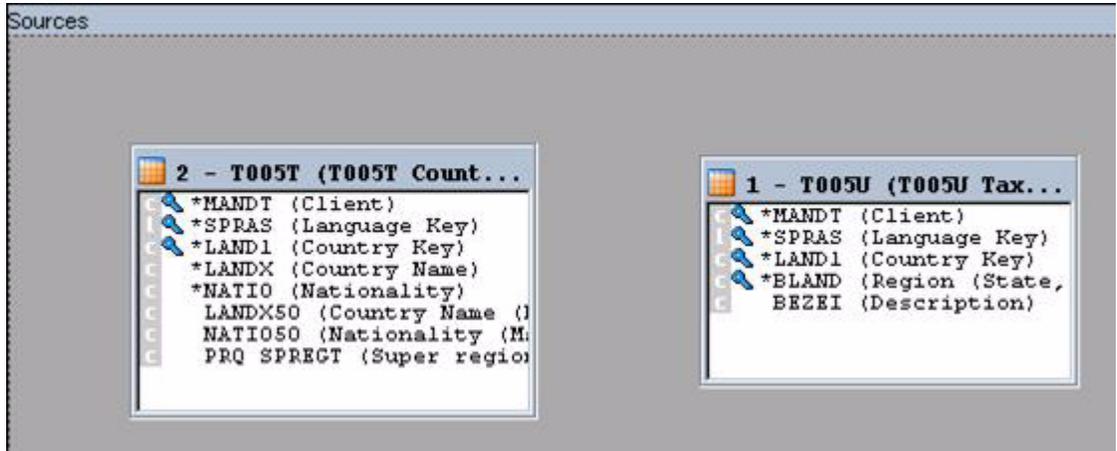
- In the Models tree view, select the WS\_GEO\_DS datastore from the Oracle Target Model.
- Drag this datastore into the Target Datastore panel (right area of the Diagram tab). This panel appears now as shown below:



The Target Datastore section of the interface diagram tab is shown. A graphic indicates the table W\_GEO\_DS is open. Below are listed five rows in this table, including Index, Name, and Mapping columns.

\*\*\*\*\*

- In the Models tree view, select the T005T Country Names datastore from the SAP ERP Source model.
- Drag this datastore into the Sources diagram panel (left area of the diagram tab).
- If a popup window prompting "Do you want to perform Automatic Mapping?" appears, click **No**.
- Perform the same operation for the T005U Taxes: Region Key datastore from the SAP ERP Source model. The Sources diagram should look as shown below:



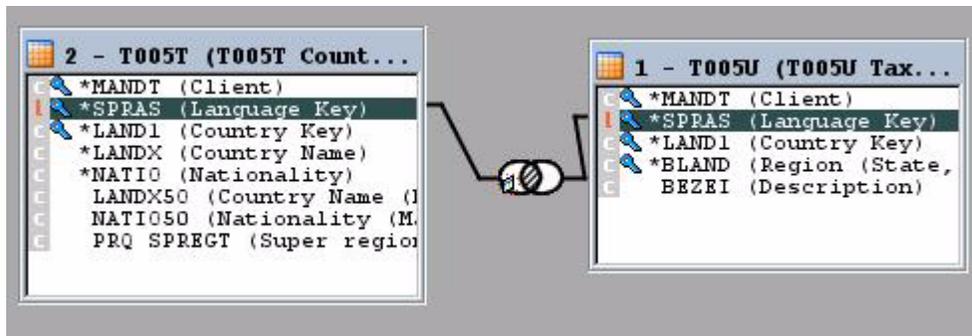
The Sources section of the interface diagram tab is shown. Two boxes represent tables that have been dragged into the Sources section. Each table shows an abbreviated table name, and a window listing table rows, with icons for key/index rows.

\*\*\*\*\*

### Define Joins between Sources

To create joins between the source datastores of an interface:

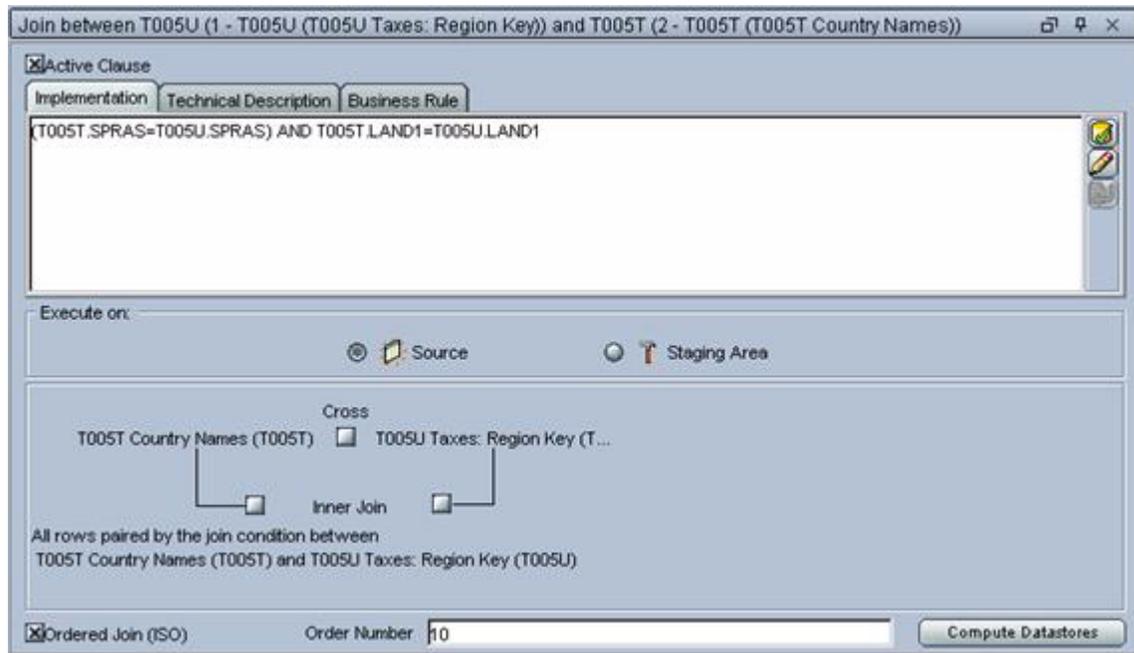
1. In the Sources diagram drag the SPRAS column from the T005T Country Name datastore onto the SPRAS column in the T005U Taxes: Region Key table. This defines the first join on the Language Key identifier, as shown below:



The Sources section of the interface diagram tab is shown. Two boxes represent tables that have been dragged into the Sources section. Each table has one row highlighted, and a line has been drawn from each highlighted row to a Union symbol that floats between the two table boxes.

\*\*\*\*\*

2. Drag the LAND1 column from the T005T Country Name table onto the LAND1 column in the T005U Taxes: Region Key table. The join is extended with the new columns. In the Properties panel, you can see the join clause:  
(T005T.SPRAS=T005U.SPRAS) AND T005T.LAND1=T005U.LAND1
3. In the Properties panel, select Ordered join and make sure that this join is executed on the **Source** as shown below:



The Properties section of the interface diagram tab is shown. At the top of the section, text reads "Join between T005U (1 - T005U (T005U Taxes: Region Key)) and T005T (2 - T005T (T005T Country Names))". The Active Clause check box is checked. The Implementation tab is selected, showing a text area with the following code: `(T005T.SPRAS=T005U.SPRAS) AND T005T.LAND1=T005U.LAND1`. In the Execute on: section, Source is selected and Staging Area is deselected. A lower area of the panel displays a graphic representation of the two objects with an inner join. At the bottom left of the panel, the check box "Ordered Join (ISO)" is checked. In the bottom center, the Order Number field has the value "10". On the bottom right is a button labeled Compute Datastores.

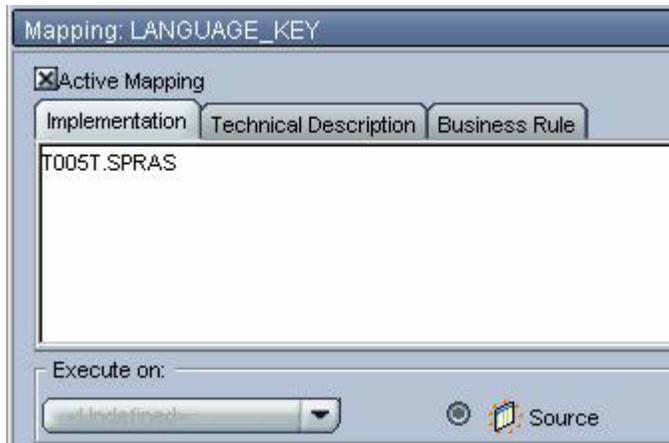
\*\*\*\*\*

4. Make sure that table number of T005U is smaller than that of T005T. In *1:n relationships* the parent table has to be the first in a join.

## Create the Mappings

To create the mappings of the target datastore:

1. Select the LANGUAGE KEY column from the target datastore.
2. Drag the SPRAS column from the T005T Country Names source table into the Implementation field of the Properties panel.
3. Make sure that the **Execution On** is set to **Source**. The mapping should look as shown below:



The Mapping: LANGUAGE\_KEY window is shown. At the top, the Active Mapping check box is checked. The Implementation tab is selected, and the field has the value "T005T.SPRAS".

\*\*\*\*\*

4. Repeat this operation to perform the following simple mappings:

- W\_GEO\_DS.COUNTRY = T005T.LAND1
- W\_GEO\_DS.COUNTY = T005T.LANDX
- W\_GEO\_DS.STATE\_PROV = T005U.BLAND
- W\_GEO\_DS.NATIONALITY = T005T.NATIO

## Define the Interface Flow

To define the interface flow:

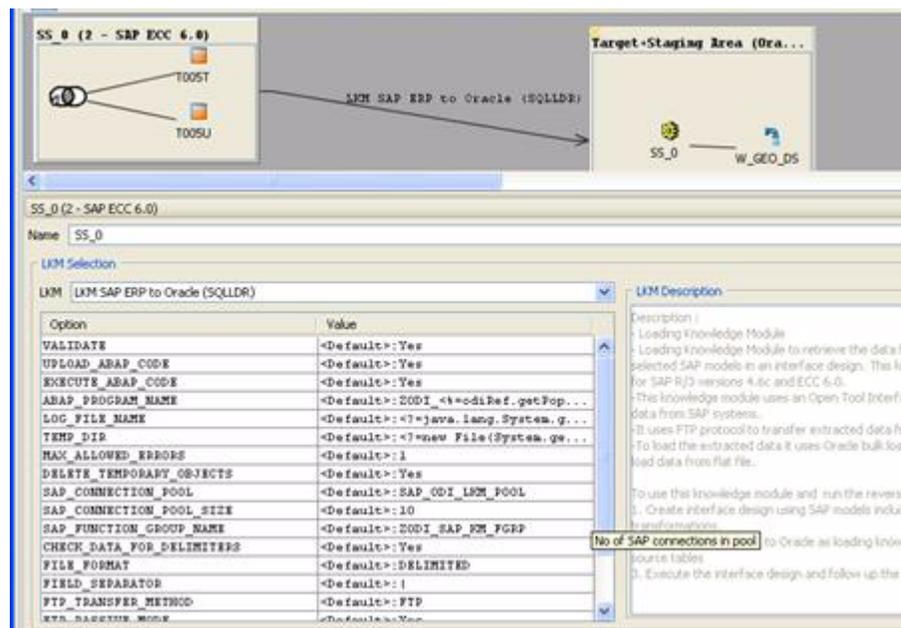
1. Select the Flow tab. The Flow diagram looks as shown below;



The Flow diagram from the Flow tab is shown. Two boxes represent the source and target data stores. In the source box, a union graphic has two lines leading to icons representing each table. A line connects the source box to the target box. In the target box, a gear icon is labeled "SS\_0" and is connected to a table icon representing the target table.

\*\*\*\*\*

2. Select the source set `SS_0` (`SAP_ERP`) that contains both the `T005T` and `T005U` tables. This source set represents the source dataset made up of two joined SAP ERP tables.
3. In the LKM Selection box, select `LKM SAP ERP to Oracle (SQLLDR)`.
4. Select the **Target + Staging Area** box. This represents the target Oracle database, which is also used as the staging area for this interface.
5. In the LKM Selection, select `LKM SAP ERP to Oracle (SQLLDR)` or `LKM SAP ERP to SQL`.
6. In the IKM Selection, select `IKM Oracle Incremental Update`.
7. Set the IKM options as follows:
  1. Select `No` for the `FLOW_CONTROL` option.
  2. If you are using FTP for the data transfer, please set the `FTP_TRANSFER_METHOD` option to `FTP`. If the FTP server is installed on your ODI agent machine, select `NONE`.
  3. Leave other options as is. The flow should look as shown below:



A portion of the flow tab is shown, with the flow diagram and the LKM Selection details displayed. The source and target boxes in the flow diagram are connected with a line labeled "LKM SAP ERP to Oracle (SQLLDR)". In the LKM Selection details, this LKM is selected and many Option:Value pairs are listed in a table.

\*\*\*\*\*

4. Click **OK** to save and close your interface.

## Running the Integration Interface

This section contains the following topics:

- [Running the Interface](#)
- [Review the Interface Execution](#)

- [Review the Resulting Data](#)

## Running the Interface

To run the integration interface:

1. In the Projects tree view, expand the SAP Demo Project
2. Expand the SDE\_SAP\_GeoDimension folder.
3. Expand the Interfaces node.
4. Select the SDE\_SAP\_GeoDimension.W\_GEO\_DS interface.
5. Right-click and select **Execute**.
6. In the Execution window that appears, click **OK**.
7. Click **OK** in the Session Started window.

### What happens when you run the interface?

First, Oracle Data Integrator creates an ABAP program that does the following:

- It performs the extraction of the data on the SAP engine. In this example, the extraction joins the two source tables and then returns the joined resultset in an extraction file.
- It uploads the extraction file into a FTP data server. This FTP host is specified via the File Logical Schema called *File Server for SAP ABAP*. This Logical Schema is mapped to a Physical Schema and therefore a data server in the given context. This data server contains the connection information for the FTP host.

Then, the ABAP code is uploaded using the *OdiSapAbapExecute* tool. The ABAP program is pushed into the SAP Function group given in the SAP\_FUNCTION\_GROUP\_NAME LKM option. This phase can be suppressed by setting the UPLOAD\_ABAP\_CODE LKM option to No.

The ABAP code is executed also by the *OdiSapAbapExecute* tool. At the end of the ABAP code execution, the extraction file is available in the FTP host.

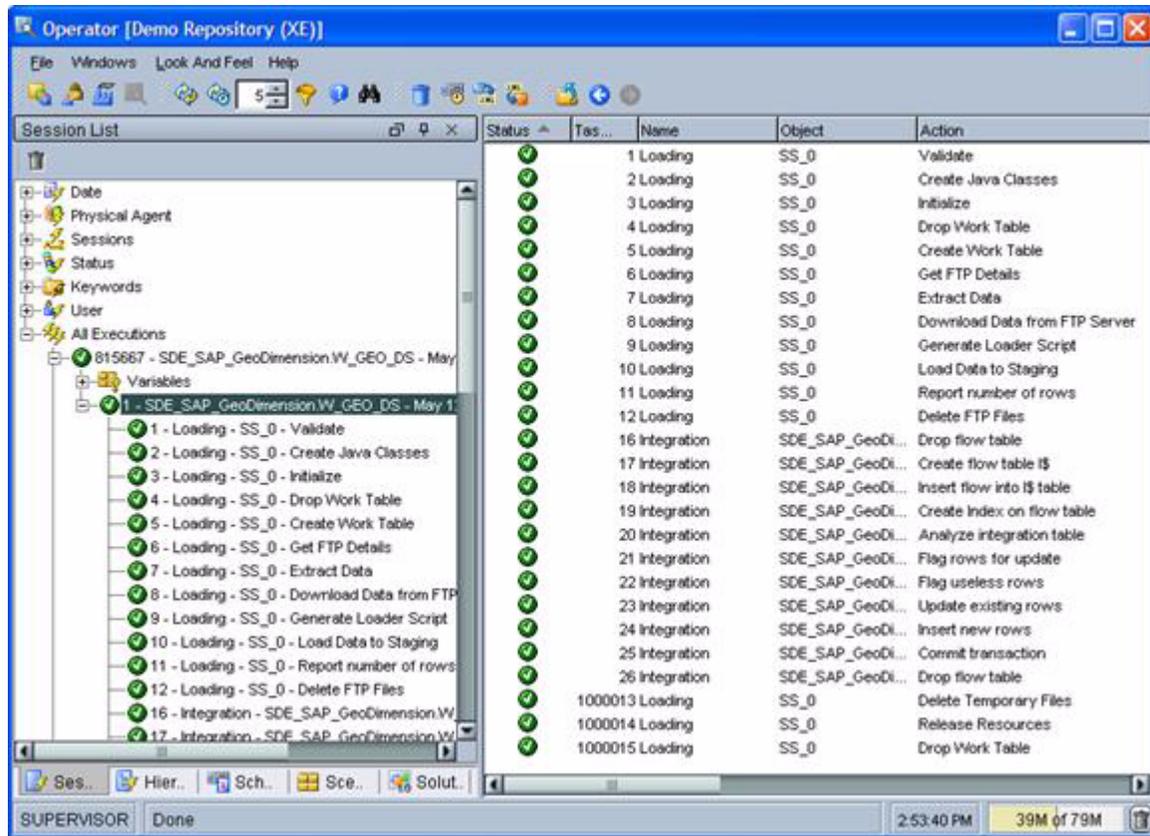
The Oracle Data Integrator agent is able to download this extraction file from the FTP host, or directly access it, depending on the FTP\_TRANSFER\_METHOD specified in the IKM option.

Finally, when the agent accesses the extraction file, it uses SQL\*Loader or a JDBC Connection to load this file into the Oracle/non-Oracle staging area. The rest of the integration process takes place within the Oracle/non-Oracle engine.

## Review the Interface Execution

To review the interface execution:

1. Connect to Operator.
2. In the Operator, select the Session List tree view.
3. Expand the All Executions node in this tree view.
4. The latest session is the first entry of this list and appears as shown below.



The Operator interface is shown with the All Executions node expanded in the Session List tree view. On the right, a table displays many rows, with the column headings Status, Task, Name, Object, and Action.

\*\*\*\*\*

5. Double click the SDE\_SAP\_GeoDimension.W\_GEO\_DS node to see the session details and the number of lines processed.

## Review the Resulting Data

To review the resulting data:

1. In the Designer, in the Models tree view, expand the Oracle Target Model.
2. Select the W\_GEO\_DS datastore, right-click and select **Data** to view the data integrated into the target Oracle table.



---

---

## Going Further with Oracle Data Integrator

This chapter describes what steps to take next with Oracle Data Integrator.

You have now completed a project extracting data from a SAP ERP system!

In this project, you have:

- Set up the environment and topology to work with SAP ERP.
- Created and reverse-engineered a SAP ERP data model
- Created an interface to load the data from several joined SAP ERP tables into the Oracle database.

Refer to the *Oracle Fusion Middleware Knowledge Module Developer's Guide for Oracle Data Integrator* for more information on the SAP ERP KM features and options.

---

---

**Note:** In case the execution did not complete successfully, please check the Operator details for the error message. In addition to this, the following log files will contain execution information.

- <System Temp Dir>/ODI\_<Interface Id>\_<SrcSet>.genlog
- <System Temp Dir>SAPAbapExecuteOpenTool\_<Interface Id>.log
- <System Temp Dir or local FTP dir>/ ZODI\_<Interface Id>\_<SrcSet>\_<Context>.log
- <System Temp Dir or local FTP dir>/ ZODI\_<Interface Id>\_<SrcSet>\_<Context>.out
- <System Temp Dir or local FTP dir>/ ZODI\_<Interface Id>\_<SrcSet>\_<Context>.err

Refer to section 8.7.4, "Log Files," in *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator* for more details.

---

---

### Going further with Oracle Data Integrator

Use the demonstration environment to familiarize yourself with Oracle Data Integrator. You can go further with Oracle Data Integrator by taking advantage of the samples available on the Oracle Technology Network.



---



---

## SAP ABAP ERP Required Privileges

This appendix lists the required privileges for accessing the SAP system using Oracle Data Integrator.

An SAP dialog user is required for connecting to the SAP system, for the RKM installation and for the LKM execution during development. A SAP RFC user is *not* sufficient.

During development, this user must have a developer license key. License keys can be retrieved from <http://service.sap.com>. Any execution attempts without this developer license key will lead to failure and may cause the need for clean-up operations.

For RKM and LKM execution in production, an SAP RFC user is sufficient.

These SAP user types can be used for the following operations:

**Table A-1 Required SAP User Types**

| Operation                | Required SAP User Type          |
|--------------------------|---------------------------------|
| RKM: Setup               | SAP Dialog user                 |
| RKM: Reverse Engineering | SAP Dialog user or SAP RFC user |
| LKM: Development         | SAP Dialog user or SAP RFC user |
| LKM: Production          | SAP Dialog user or SAP RFC user |

The following tables list the privileges required for using SAP ABAP Knowledge Modules:

- [Authorizations Required for RKM SAP ERP Setup](#)
- [Authorizations Required for RKM Execution](#)
- [Authorizations Required for LKM Execution](#)
- [Authorizations Required for LKM Execution for Production](#)

### Authorizations Required for RKM SAP ERP Setup

The following authorizations are required for setting up the configuration for the **RKM SAP ERP** and install the required RFC programs to retrieve the metadata about SAP tables.

---



---

**Note:** Developer key is required in this authorization.

---



---

| Object     | Field Name | Value      |
|------------|------------|------------|
| S_ADMI_FCD | S_ADMI_FCD | NADM       |
| S_ADMI_FCD | S_ADMI_FCD | PADM       |
| S_ADMI_FCD | S_ADMI_FCD | SM21       |
| S_DATASET  | ACTVT      | *          |
| S_DATASET  | FILENAME   | *          |
| S_DATASET  | PROGRAM    | *          |
| S_DEVELOP  | ACTVT      | 01, 02, 03 |
| S_DEVELOP  | DEVCLASS   | *          |
| S_DEVELOP  | OBJNAME    | *          |
| S_DEVELOP  | OBJTYPE    | *          |
| S_DEVELOP  | P_GROUP    | *          |
| S_GUI      | ACTVT      | *          |
| S_RFC      | ACTVT      | *          |
| S_RFC      | RFC_NAME   | *          |
| S_RFC      | RFC_TYPE   | *          |
| S_RZL_ADM  | ACTVT      | *          |
| S_TABU_DIS | ACTVT      | 02, 03     |
| S_TABU_DIS | DICBERCLS  | *          |
| S_TCODE    | TCD        | AL11       |
| S_TCODE    | TCD        | SE10       |
| S_TCODE    | TCD        | SE11       |
| S_TCODE    | TCD        | SE16       |
| S_TCODE    | TCD        | SE37       |
| S_TCODE    | TCD        | SE38       |
| S_TCODE    | TCD        | SE38       |
| S_TCODE    | TCD        | SE80       |
| S_TCODE    | TCD        | SE80       |
| S_TCODE    | TCD        | SE91       |
| S_TCODE    | TCD        | SM21       |
| S_TCODE    | TCD        | SM50       |
| S_TCODE    | TCD        | SM58       |
| S_TCODE    | TCD        | SM59       |
| S_TCODE    | TCD        | ST22       |
| S_TCODE    | TCD        | SU53       |
| S_TCODE    | TCD        | SU53       |
| S_TRANSPRT | ACTVT      | 01, 02, 03 |
| S_TRANSPRT | TTYPE      | *          |

## Authorizations Required for RKM Execution

The following authorizations are required for running a reverse-engineering using the RKM SAP ERP.

| Object     | Field Name | Value |
|------------|------------|-------|
| S_RFC      | ACTVT      | 16    |
| S_RFC      | RFC_NAME   | *     |
| S_RFC      | RFC_TYPE   | *     |
| S_TABU_DIS | ACTVT      | 03    |
| S_TABU_DIS | DICBERCLS  | *     |
| S_TCODE    | TCD        | SE16  |

## Authorizations Required for LKM Execution

The following authorizations are required for using the LKM SAP ERP to Oracle (SQLLDR). This set of authorization is needed to install the ABAP code generated by the LKM and execute it.

**Note:** Developer key is required in this authorization

| Object     | Field Name | Value      |
|------------|------------|------------|
| S_DATASET  | ACTVT      | 06, 33, 34 |
| S_DATASET  | FILENAME   | *          |
| S_DATASET  | PROGRAM    | *ZODI*     |
| S_DEVELOP  | ACTVT      | 01, 02, 03 |
| S_DEVELOP  | DEVCLASS   | *          |
| S_DEVELOP  | OBJNAME    | *          |
| S_DEVELOP  | OBJTYPE    | *          |
| S_DEVELOP  | P_GROUP    | *          |
| S_RFC      | ACTVT      | *          |
| S_RFC      | RFC_NAME   | *          |
| S_RFC      | RFC_TYPE   | *          |
| S_TRANSPRT | ACTVT      | *          |
| S_TRANSPRT | TTYPE      | *          |

## Authorizations Required for LKM Execution for Production

The following authorizations are required for using the LKM SAP ERP to Oracle (SQLLDR) in a production environment. In such environment, the ABAP code generated by the LKM is not installed by the LKM but simply executed.

| Object     | Field Name | Value |
|------------|------------|-------|
| S_CTS_ADMI | CTS_ADMFCT | *     |

| <b>Object</b> | <b>Field Name</b> | <b>Value</b> |
|---------------|-------------------|--------------|
| S_DATASET     | ACTVT             | *            |
| S_DATASET     | FILENAME          | *            |
| S_DATASET     | PROGRAM           | *ZODI*       |
| S_RFC         | ACTVT             | *            |
| S_RFC         | RFC_NAME          | *            |
| S_RFC         | RFC_TYPE          | *            |

---

---

## Installing the librfc32/librfc64 Library

This appendix describes how to install the librfc32 or librfc64 library.

The SAP JCo software requires the librfc32/librfc64 library, but does not include it. It is included for example with the SAP GUI software. In case this library is not yet installed, this section describes how to download and install it. These steps are only required, if librfc32/librfc64 is not yet installed.

### Installing the librfc32/librfc64 Library

Perform the following steps to install the librfc32/librfc64 library:

1. Connect to SAP Support Portal <http://service.sap.com>.
2. Go to SAP support portal.
3. Go to Software Downloads.
4. Go to the SAP Software Download Center.
5. Select Support Packages and Patches.
6. Select Browse Our Download Catalog.
7. Select Additional Components.
8. Download appropriate RFCSDK version (SAP NW RFC SDK, SAP RFC SDK, SAP RFC SDK UNICODE)
9. Use SAPCAR extractor to extract the content of the downloaded SAR file
10. For JCo installed on Windows:

Copy the file `librfc[u][32|64].dll` located in folder `rfcsdk/bin` to `c:\windows\system32` (exact folder name depends on windows version).

For JCo installed on Unix:

Copy the file `librfccm[u].so` to the folder containing the `sapjco3.jar` and `libsapjco3.so`. This directory may need to be added to `LD_LIBRARY_PATH`.



## SAP Stand-Alone Connection Test

This appendix describes how to perform an SAP stand-alone connection test.

In addition to the Connection Testing described above, a test can be performed outside of ODI using a standalone java utility.

### SAP Stand-Alone Connection Test

To use a standalone java utility to test the connection:

1. Open a command window.
2. Change to oracledi/drivers directory.
3. Make sure that JAVA\_HOME points to a JVM (at least 1.5).
4. Make sure that you have installed SAP Java Connector and that the sapjco3.jar and the sapjco3 library are in the oracledi/drivers directory.
5. Launch the utility using the following command:

On Windows:

```
java -cp sapjco3.jar;odi-sap.jar oracle.odi.sap.km.test.JCoTest
```

On Linux/UNIX:

```
java -cp sapjco3.jar:odi-sap.jar oracle.odi.sap.km.test.JCoTest
```

This should result in an output similar to this:

```
JCO3 Library available!!
JCO Version: 3.0.2 (2009-03-29)
Creating connection Pool...
Error: com.sap.conn.jco.JCoException: (102) RFC_ERROR_COMMUNICATION: Connect to SAP gateway failed
Connection parameters: TYPE=A DEST=ODI_SAP_CON_POOL ASHOST=<SAP Application Server> SYSNR=00 PCS=1
LOCATION      CPIC (TCP/IP) on local host with Unicode
ERROR        hostname '<SAP Application Server>' unknown
TIME         Tue Sep 08 16:07:27 2009
RELEASE      711
COMPONENT    NI (network interface)
VERSION      39
RC           -2
MODULE       ninti.c
LINE         895
DETAIL       NIPGetHostByName: '<SAP Application Server>' not found
SYSTEM CALL  getaddrinfo
COUNTER      2
```

A command-line console is shown with text output. The output includes an error message "Error: com.sap.conn.jco.JCoException: (102) RFC\_ERROR\_COMMUNICATION: Connect to SAP gateway failed", with connection parameters. Following is a list of details including location, error, time, release number, component, version, RC, module, line, detail, system call, and counter.

```
*****
```

6. Use a text editor to open the `ODI_SAP_CON_POOL.jcoDestination` file in the `oracledi/drivers` directory. The file should look like this:

```
#for tests only!  
jco.client.lang=EN  
jco.destination.peak_limit=10  
jco.client.client=800  
jco.client.passwd=<SAP Password>  
jco.client.user=<SAP User>  
jco.client.sysnr=00  
jco.destination.pool_capacity=5  
jco.client.ashost=<SAP Application Server>
```

7. Enter your SAP connection information, which you have received from your SAP administrator. The file will then look similar to this (use your connection information!):

```
#for tests only !  
jco.client.lang=EN  
jco.destination.peak_limit=10  
jco.client.client=800  
jco.client.passwd=ODI123  
jco.client.user=ODI  
jco.client.sysnr=00  
jco.destination.pool_capacity=5  
jco.client.ashost=123.123.123.123.
```

8. Launch the utility again using the following command:

On Windows:

```
java -cp sapjco3.jar;odi-sap.jar oracle.odi.sap.km.test.JCoTest
```

On Linux/UNIX:

```
java -cp sapjco3.jar:odi-sap.jar oracle.odi.sap.km.test.JCoTest
```

This should result in an output similar to this:

```

C:\WINDOWS\system32\cmd.exe
SYSTEM CALL getaddrinfo
COUNTER      2

D:\OraHome_1\oracledi\drivers>java -cp odi-sap.jar;sapjco3.jar oracle.odi.sap.kn.test.JCoTest
JCO3 Libraray available!!
JCO Version: 3.0.2 (2009-03-29)
Creating connection Pool...
R3 Name: null
Attributes: DEST: ODI_SAP_CON_POOL
OWN_HOST:
PARTNER_HOST:
SYSTNR: 00
SYSID: NRI
CLIENT: 800
USER:
LANGUAGE: E
ISO_LANGUAGE: EN
OWN_CODEPAGE: 1100
OWN_CHARSET: ISO8859_1
OWN_ENCODING: iso-8859-1
OWN_BYTES_PER_CHAR: 1
PARTNER_CODEPAGE: 1100
PARTNER_CHARSET: ISO8859_1
PARTNER_ENCODING: iso-8859-1
PARTNER_BYTES_PER_CHAR: 1
OWN_REL: 711
PARTNER_REL: 46C
PARTNER_TYPE: 3
KERNEL_REL: 46D
TRACE:
RFC_ROLE: C
OWN_TYPE: E
CPIC_CONUID: 00000000

Successfully Connected to SAP System...
Z_ODI RFC GET TABLES is exist in specified SAP System...
Z_ODI RFC READ TABLE is exist in specified SAP System...
Z_ODI RFC TABLE KEYS is exist in specified SAP System...
Z_ODI RFC GET TABLE INDEXES is exist in specified SAP System...

D:\OraHome_1\oracledi\drivers>

```

Command-line output results from running the java command in this step is shown. The console reports creating a connection pool, lists connection pool attributes such as host, client, IDs, language, and so forth; and then reports "Successfully Connected to SAP system..." with lines reporting the existence of four Function Modules in the specified SAP System.

\*\*\*\*\*

In addition to just testing the SAP connection, the utility will also validate the existence of certain Function Modules required for the RKM. These are installed during first execution of the RKM (UPLOAD\_ABAP = Yes).

9. Delete the ODI\_SAP\_CON\_POOL.jcoDestination file after execution, as it contains the SAP login credentials.



---

---

## Uninstalling ODI SAP Components

This appendix describes how to uninstall ODI components from your SAP system.

In case of upgrading to a newer ODI SAP Connector release or for permanently removing ODI connectivity, please follow the steps described in this section.

Please note that this will delete ALL components including generated extractor programs.

### Uninstalling ODI SAP Components

To upgrade to a newer ODI SAP Connector release or to permanently remove the ODI connectivity:

1. Start the SAP GUI.
2. Connect to the SAP systems you want to uninstall.
3. Go to the transaction SE80.
4. Select the package or development class option from the drop down box.
5. Enter ZODI\_DEVCLASS in the package or development class field.
6. Right-click the ZODI\_DEVCLASS object below the object name tab.
7. Select **Delete** in the drop down menu and delete the development class.  
If it does not allow deleting the development class individually, delete all the objects one by one and then delete the development class.
8. Go to transaction SE10. Select **Modifiable** and **Workbench Requests**. Click **Display**.
9. Select the transport requests which have the ODI Objects and Release them.
10. Now repeat the steps 3 to 7 to delete Development Class.
11. Go to transaction SE01. Select **Modifiable** and **Workbench Requests**. Click **Display**. Release the Transport Request which has ZODI\_DEVCLASS.

