

Oracle Warehouse Builder 10g: Implementation Part 1

Volume II • Student Guide

D18398GC10

Edition 1.0

June 2006

D46579

ORACLE®

Authors

Molly Correa
Richard Green

**Technical Contributors
and Reviewers**

David Allan
Sharath Bhujani
Maria Billings
Michelle Bird
Herbert Bradbury
Dairy Chan
Jean-Pierre Dijcks
Nancy Greenberg
Joe Klein
David Last
John Leigh
Ben Loper
Paul Narth
Padu Potineni
John Potter
Brian Pottle
Kasturi Shekhar
Puja Singh
Dick Whittington
Frank Yang
OWB Professional
Community, EMEA &
Worldwide (via Jon
Ainsworth, Tony
Carpenter)

Publisher

Sujatha Nagendra

Editors

Aju Kumar
Navratan Singh

Graphic Specialist
Samir Mozumdar

Copyright © 2006, Oracle. All rights reserved.

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

Trademark Notice

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

vivek.kamra@rbs.com
use this Student Guide
for a non-transferable license to

Contents

Preface

I Introduction

Lesson Objectives	I-2
Course Objectives	I-3
Agenda: Day 1	I-4
Agenda: Day 2	I-5
Agenda: Day 3	I-6
Appendices	I-7
Questions About You	I-8
Summary	I-9

1 Setting Up and Starting Warehouse Builder

Objectives	1-2
What Is Oracle Warehouse Builder?	1-3
Producing Quality Information	1-4
New Segmentation of OWB Functionality	1-5
BI Environment from an OWB Perspective	1-7
OWB Sources and Targets	1-8
Installing OWB	1-9
Hardware and Software Requirements	1-10
Supported and Certified Versions of Oracle RDBMS	1-11
Software Versions Compatible with Warehouse Builder	1-12
OWB Product Components	1-13
Installing and Configuring OWB	1-14
Configuring OWB	1-15
Classroom Stand-alone Configuration	1-16
OWB 10g R2 Default Installation with a Single Repository	1-17
Target Users	1-18
Logging In to OWB Design Center	1-19
Practice 1-1: Creating a Repository Owner and User	1-20
Projects	1-21
Organizing Your Work in Projects	1-22
Concurrent Users per Project with Object Locking	1-23
Multiple User Access	1-24

Design Center 1-25
Objects Within a Project 1-26
Objects Within a Module 1-31
Locations, Connectors, and Control Centers 1-35
Global Explorer Objects 1-37
Creating a Project 1-38
Setting Project Preferences 1-39
Summary 1-41
Practice 1-2: Invoking OWB and Opening a Project 1-42

2 Introducing OWB 10g R2 Architecture and Configuration

Objectives 2-2
Process Flow of Design and Deployment 2-3
OWB 10g R2 Repository Default Installation 2-5
OWB 10g R2 Repository: Design Phase 2-6
OWB Design Components 2-7
OWB 10g R2 Repository: Run Time 2-9
OWB Run-Time Components 2-10
Loose Coupling of Metadata with Physical Implementations 2-12
Multiple Named Configurations: Preview 2-13
OWB Mapping 2-14
Executing a Mapping at a Target Location 2-15
Classroom Stand-alone Configuration 2-16
Independent Versus Dependent Marts 2-17
Independent Data Mart 2-18
Dependent Data Mart 2-19
Purpose of a Staging Area 2-20
Data Staging Area 2-21
Remote Staging Model 2-22
On-Site Staging Model 2-23
Purpose of an Enterprise Model 2-24
Course: Predefined and New Objects 2-25
OMB*Plus tcl Scripting Within Design Client 2-26
Launching OMB*Plus from the Command Line 2-27
Summary 2-28
Practice 2-1: Importing Metadata 2-29

3 Defining Source Metadata

Objectives 3-2
Course Progress Report 3-3
Metadata to Data 3-4

Source Module	3-5
Lesson Data Sources	3-6
Creating a Flat File Source Module	3-7
Using the Create Module Wizard	3-8
Create Module Wizard: Name and Description	3-9
Create Module Wizard: Connection Information	3-10
Editing File System Location	3-11
Create Module Wizard: Summary	3-12
Metadata to Data	3-13
Import Metadata Wizard: Welcome	3-14
Importing from Flat Files	3-15
Selecting the Flat Files	3-16
Collecting Geography Metadata Information	3-17
Flat File Sample Wizard: Welcome	3-18
Describing the File	3-19
Record Organization	3-20
Selecting the File Format	3-22
File Layout	3-23
Modifying Field Information	3-24
Summary for Geography Metadata Sampling	3-25
Practice 3-1: Create a Flat File Source Module	3-26
Lesson Data Sources	3-27
Creating an Oracle Source Module	3-28
Create Module Wizard: Name and Description	3-29
Create Module Wizard: Connection Information	3-30
Editing the Oracle Database Location	3-31
Finishing Module Creation and Proceeding to Import	3-32
Import Metadata Wizard: Filter Information	3-33
Selecting the Tables for Import	3-34
Completing the Metadata Import	3-35
Examining the Imported Metadata	3-36
Viewing Data Using the Data Viewer	3-37
Difference Between Obtaining Relational and Flat File Source Metadata	3-38
Summary	3-39
Practice 3-2: Create a Relational Database Source Module	3-40

4 Ensuring Data Quality Using Data Profiling

Objectives	4-2
Course Progress Report	4-3
What Is Data Profiling?	4-4
Steps in Data Profiling	4-5

Creating a Data Profile 4-6
Create Data Profile Wizard: Name and Description 4-7
Create Data Profile Wizard: Select Objects 4-8
Create Data Profile Wizard: Summary 4-9
Steps in Data Profiling 4-10
Profiling the Data Using the Data Profile Editor 4-11
Data Profile Setup 4-12
Preparing the Profile 4-13
Profiling Initiated 4-14
Retrieving Profile Results 4-15
Practice 4-1: Profile the Data 4-16
Steps in Data Profiling 4-17
Types of Profiling 4-18
Attribute Analysis 4-19
Attribute Analysis: Glimpse of the Table Data 4-20
Attribute Aggregation Analysis 4-21
Aggregation Graphical Results 4-22
Determining the Six-Sigma Score 4-23
Attribute Data Type Analysis 4-24
Attribute Domain Analysis 4-25
Attribute Unique Key Analysis 4-26
Referential Analysis 4-27
Property Inspector 4-28
Performing Referential Analysis 4-29
Referential Analysis: Noncompliant 4-30
Joins Subtab on the Referential Page 4-31
Functional Dependency Analysis 4-32
Performing Functional Dependency Analysis 4-33
Summary 4-34
Practice 4-2: View and Analyze the Profile Results 4-35

5 Defining Staging Metadata and Mapping Tables

Objectives 5-2
Purpose of a Staging Area 5-3
Defining a Staging Area Warehouse Module 5-4
Course Progress Report 5-5
Creating the Target User 5-6
Create User Wizard: Select DB User to Register 5-7
Create User Wizard: Check User as Target Schema 5-8
Creating the Warehouse Target Module 5-9
Create Module Wizard: Name and Description 5-10

Create Module Wizard: Connection Information	5-11
What Are External Tables?	5-12
Creating an External Table	5-13
Create External Table: Name and Description	5-14
Create External Table: File Selection	5-15
Create External Table: Locations	5-16
Configuring External Tables	5-17
Practice 5-1: Overview	5-18
Practice 5-1: Create an External Table	5-19
Metadata to Data	5-21
Building a Mapping	5-22
Designing the LOAD_STG_GEOG Mapping	5-23
OWB Mapping Editor	5-24
Mapping Editor User Interface	5-25
Building a Mapping	5-26
Choosing the External Table	5-27
Creating a Staging Table	5-28
Adding an Attribute	5-29
Using a Constant Operator	5-30
Using the Expression Builder to Define a Constant Operator	5-31
Using a Filter Operator	5-32
Using the Expression Builder to Define a Filter Condition	5-33
Using a Joiner	5-34
Key Lookup Operator	5-35
Key Lookup Condition	5-36
Set Operation Operator	5-37
Building a Mapping	5-38
Mapping Source Columns to the Target	5-39
Using the Create and Bind Option to Create the Repository Table	5-40
Setting Load Type to TRUNCATE/INSERT	5-41
Synchronizing Operators and Repository Objects	5-42
Building a Mapping	5-43
Validating the Mapping	5-44
Generating the Mapping	5-45
Building a Mapping	5-46
Deploying a Mapping	5-47
Building a Mapping	5-48
Executing a Mapping	5-49
Metadata to Data	5-50
Viewing the Data Using Data Viewer	5-51
Pluggable Mappings	5-52

Advantages of Pluggable Mappings	5-53
Example of a Pluggable Mapping	5-54
Target Load Ordering	5-55
Overriding the Default Load Order	5-56
Summary	5-57
Practice 5-2: Create a Simple Mapping from Source Objects to the Target	5-58

6 Deriving Data Rules and Running Correction Mappings

Objectives	6-2
Course Progress Report	6-3
Lesson Progress Report	6-4
Data Rules	6-5
Deriving Data Rules	6-6
Derive Data Rule Wizard: Welcome	6-7
Derive Data Rule Wizard: Name and Description	6-8
Derive Data Rule Wizard: Define Rule	6-9
Derive Data Rule Wizard: Summary	6-10
Examining the Data Rule Panel	6-11
Lesson Progress Report	6-12
Creating Corrections Using the Create Correction Wizard	6-13
Create Correction Wizard: Select Target Module	6-14
Create Correction Wizard: Select Objects	6-15
Create Correction Wizard: Select Data Rules	6-16
Create Correction Wizard: Verify and Accept Corrected Tables	6-17
Create Correction Wizard: Constraints Tab	6-18
Create Correction Wizard: Choose Data Correction Actions	6-19
Lesson Progress Report	6-20
Examining the Correction Module	6-21
Examining the Generated Correction Mapping	6-22
Examining the Transformations Node	6-24
Summary	6-25
Practice 6-1: Deriving Data Rules and Creating Correction Mappings	6-26

7 Defining a Relational Dimensional Model

Objectives	7-2
Course Progress Report	7-3
Lesson Scenario	7-4
Classifying the Data	7-5
Dimension Tables	7-6
Determining Granularity	7-7
Dimensional Design Using OWB	7-8

Defining Dimensions	7-9
Creating a Dimension Using the Wizard	7-10
Create Dimension Wizard: Name and Description	7-11
Create Dimension Wizard: Storage Type	7-12
Create Dimension Wizard: Dimension Attributes	7-13
Create Dimension Wizard: Levels	7-14
Create Dimension Wizard: Level Attributes	7-15
Create Dimension Wizard: Data Policy	7-16
Create Dimension Wizard: Precreate Settings	7-17
Create Dimension Wizard: Dimension Creation Progress	7-18
Configuring a Dimension	7-19
Practice 7-1: Create a Dimension Using the Wizard	7-20
About the Data Object Editor	7-21
Detail View	7-22
Creating or Editing a Dimension Using the Data Object Editor	7-23
Creating a Dimension Using the Editor	7-24
Dimension Details: Name Tab	7-25
Creating a Dimension Role	7-26
Dimension Details: Storage Tab	7-27
Dimension Details: Attributes Tab	7-28
Dimension Details: Levels Tab	7-29
Dimension Details: Hierarchies Tab	7-30
Dimension Details: SCD Tab	7-31
Relational Implementation: Star	7-32
Relational Implementation: Snowflake	7-33
Binding	7-34
Autobinding	7-35
Manual Binding	7-36
Unbind	7-38
Dimension Details: Data Viewer Tab	7-39
Practice 7-2: Create a Dimension Using the Editor	7-40
Creating the Time Dimension by Using the Time Wizard	7-41
Create Time Dimension Wizard: Name and Description	7-42
Create Time Dimension Wizard: Storage	7-43
Create Time Dimension Wizard: Data Generation	7-44
Create Time Dimension Wizard: Levels	7-45
Create Time Dimension Wizard: Precreate Settings	7-46
Create Time Dimension Wizard: Progress Panel	7-47
Create Time Dimension Wizard: Summary	7-48
Practice 7-3: Create a TIMES Dimension Using the Time Wizard	7-49
Defining a Cube Using the Wizard	7-50

Create Cube Wizard: Name and Description	7-51
Create Cube Wizard: Storage Type	7-52
Create Cube Wizard: Dimensions	7-53
Create Cube Wizard: Measures	7-54
Create Cube Wizard: Summary	7-55
Editing a Cube Using the Data Object Editor	7-56
Assigning Roles	7-57
Configuring Cubes	7-58
Practice 7-4: Create a Cube Using the Editor	7-59
Designing ETL Mappings	7-60
Mapping to Load the PRODUCTS Dimension	7-61
Mapping to Load the SALES Cube	7-62
Pluggable Mapping	7-64
Generating the Mappings	7-65
Summary	7-66
Practice 7-5: Design ETL Mappings	7-67

8 Handling Slowly Changing Dimensions

Objectives	8-2
What Is a Slowly Changing Dimension?	8-3
Types of Slowly Changing Dimensions	8-4
Type 1 SCD: Does Not Store History	8-5
Type 2 SCD: Preserves Complete History	8-6
Type 3 SCD: Stores Only the Previous Value	8-7
Creating a Type 2 SCD Using the Dimension Editor	8-8
Creating a Type 2 SCD	8-9
Mapping to Load the Type 2 SCD	8-13
Creating a Type 2 SCD Using the Wizard	8-14
Dimension Operator in a Mapping	8-15
Group Properties in the Dimension	8-16
Creating a Type 3 SCD	8-17
Attributes in a Type 3 SCD	8-18
Summary	8-19
Practice 8-1: Handling Slowly Changing Dimension	8-20

9 Using Process Flows

Objectives	9-2
Course Progress Report	9-3
Process Flow Concepts	9-4
Creating a Process Flow Module	9-5
Creating a Process Flow Package	9-6

Creating a Process Flow	9-7
Process Flow Editor	9-8
Types of Activities	9-9
Adding a Fork Activity	9-10
Adding a Mapping Activity	9-11
Adding an And Activity	9-12
Adding End Activities	9-13
Creating Transitions Between Activities	9-14
Transition Conditions	9-15
Generating the Process Flow	9-16
Practice 9-1: Create the LOAD_SALES_WH Process Flow	9-17
New Concepts in Process Management	9-18
Creating Activity Templates	9-19
Process Flow Activities	9-21
Creating a Variable	9-22
Assign Activity	9-23
Data Auditor	9-24
For Loop Activity	9-25
Route and While Loop Activities	9-26
Manual Activity	9-27
Notification Activity	9-28
Set Status Activity	9-29
SQLPLUS Activity	9-30
Wait Activity	9-31
Scheduling Concepts	9-32
Creating a Schedule Module	9-33
Creating a Schedule	9-34
Create Schedule Wizard: Name and Description	9-35
Create Schedule Wizard: Start and End Time	9-36
Create Schedule Wizard: Frequency and Repeat Interval	9-37
Monthly Schedule	9-38
Editing a Schedule	9-39
Schedule Example	9-40
Associating Executable with a Schedule	9-41
Assigning a Schedule to an Executable Object	9-42
Summary	9-43
Practice 9-2: Create a Schedule	9-44

10 Deploying, Loading, and Viewing Data

Objectives	10-2
Course Progress Report	10-3

Logical Versus Physical	10-4
Physical Implementation	10-5
Configuring for Physical Implementation	10-6
Accessing Object Configuration	10-7
Deployment Concepts	10-8
Registering Locations	10-9
Unregistering Locations	10-10
Configuration and Control Center	10-11
Active Configuration	10-12
Control Center and Locations	10-13
Control Center Manager Components	10-14
Invoking the Control Center Manager	10-15
Control Center Manager Interface	10-16
Default Action and Deploy	10-17
Deploy Options	10-18
Asynchronous Deployment and Execution	10-19
Deployment Tools Preferences	10-20
Deployment Process Preferences	10-21
Deploying Objects	10-22
Deploying Sequences	10-23
Control Center Jobs	10-24
OWB Automatically Creates a Connector	10-25
Loading the Target Warehouse	10-26
Course Progress Report	10-27
Executing Mappings	10-28
Executing the Process Flow	10-29
Execution Window	10-30
Course Progress Report	10-31
Viewing Data Using the Data Viewer	10-32
Practice 10-1: Deploying, Loading, and Viewing Data	10-33
Managing Metadata Using Dependency Manager	10-34
Integrated Lineage and Impact Analysis	10-35
Invoking Lineage and Impact Analysis	10-36
Lineage Analysis	10-37
Impact Analysis	10-38
Tree View	10-39
User-Defined Objects and Icons	10-40
Creating an Icon Set	10-41
Using an Icon Set	10-42
User-Defined Objects	10-43
Repository Browser	10-44

Starting the OWB Browser Listener	10-45
Logging In to the Warehouse Builder Browser	10-46
Browsing Design Repository Reports	10-47
Browsing Control Center Reports	10-48
Types of Control Center Reports	10-49
Object Summary Report	10-50
Execution Summary Report	10-51
Summary	10-52
Practice 10-2: Metadata Management	10-53

11 Business Intelligence Integration

Objectives	11-2
Business Justification: Tools Integration	11-3
Integration with Discoverer	11-4
Two Ways to Define Discoverer Objects in Warehouse Builder	11-5
Creating a Business Definition Module	11-6
Creating a Business Area	11-7
Creating a Simple Item Folder	11-8
Creating a Drill Path	11-12
Deriving Individual Objects	11-13
Selecting Target and Rules for Derivation	11-14
Examining Derived Business Definitions	11-15
Deriving from a Collection	11-16
Creating a Complex Item Folder	11-18
Creating a List of Values	11-19
Creating Drill to Detail	11-20
Practice 11-1: Defining Business Intelligence Objects Based on Data Objects	11-21
Generating and Deploying the Intelligent Objects	11-22
Analyzing Data Using OracleBI Discoverer Components	11-23
Connecting to Discoverer Plus Relational	11-24
Examining EUL in Discoverer	11-25
Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer	11-27
Integration with BI Beans	11-28
Creating a Business Presentation Module	11-29
Creating a Presentation Template	11-30
Selecting the Report Type	11-31
Selecting the Items for the Report	11-32
Specifying the Report Layout	11-33
Connecting to the BI Beans Report Viewer	11-34
Examining the BI Beans Report	11-35

Drilling Down into the Report 11-36

Summary 11-37

Appendix A: Practices

Appendix B: Practices Solutions

Appendix C: MOLAP: Modeling and Loading

Appendix D: Create the LOAD_SALES Mapping

Appendix A

Practices

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Table of Contents

Practice 1-1: Create a Repository Owner and User	4
Practice 1-2: Invoking OWB and Opening a Project.....	8
Practice 2-1: Importing Metadata	12
Practice 3-1: Create a Flat File Source Module	16
Practice 3-2: Create a Relational Database Source Module	16
Practice 4-1: Profile the Data.....	19
Practice 4-2: View and Analyze the Profile Results	19
Practice 5-1: Create an External Table	23
Practice 5-2: Create a Simple Mapping from Source Objects to the Target.....	24
Practice 6-1: Deriving Data Rules and Creating Correction Mappings.....	27
Practice 7-1: Create a Dimension Using the Wizard	29
Practice 7-2: Create a Dimension Using the Editor	31
Practice 7-3: Create a TIMES Dimension Using the Time Wizard.....	34
Practice 7-4: Create a Cube Using the Editor	35
Practice 7-5: Design ETL Mappings.....	35
Practice 8-1: Handling Slowly Changing Dimensions	43
Practice 9-1: Create the LOAD_SALES_WH Process Flow	46
Practice 9-2: Create a Schedule	48
Practice 10-1: Deploying, Loading, and Viewing Data.....	52
Practice 10-2: Metadata Management	54
Practice 11-1: Defining Business Intelligence Objects Based on Data Objects	60
Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer.....	64

Practice for Lesson 1

This practice is to enable you to invoke Oracle BI Warehouse Builder Design Center and create a repository owner and repository user by using the Logon window.

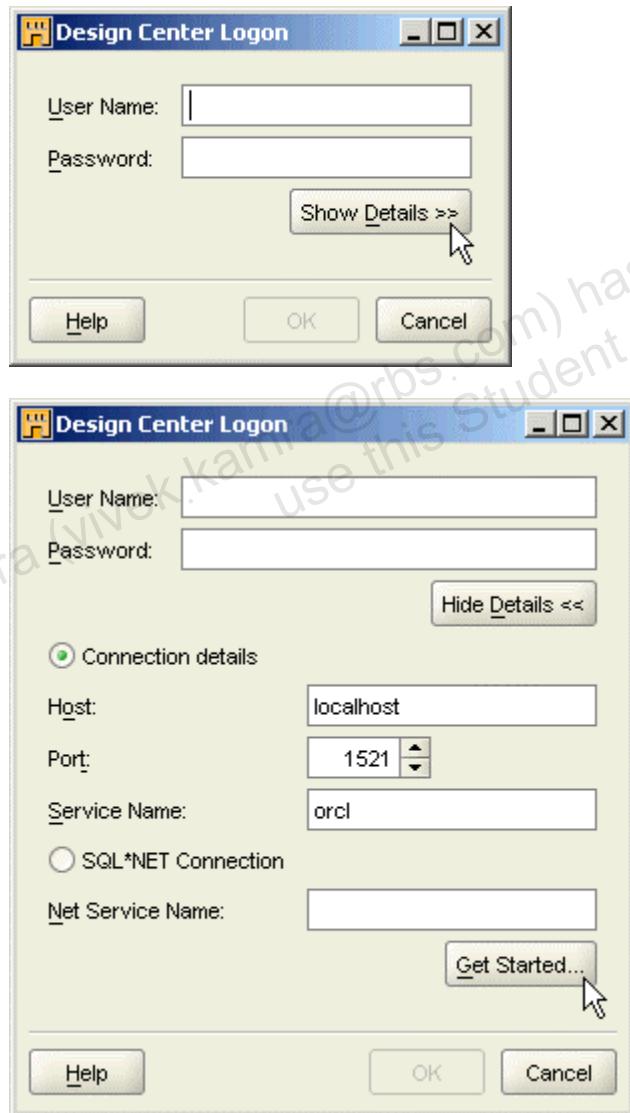
vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 1-1: Create a Repository Owner and User

Before logging in to the Design Center, you must create a user with which to log in. You can create users with the Get Started button of the Design Center Logon dialog box. You create a repository user and a highly privileged repository owner. Then you install the OWB repository.

Using the OWB Design Center Get Started Button to Create a Repository User and Owner

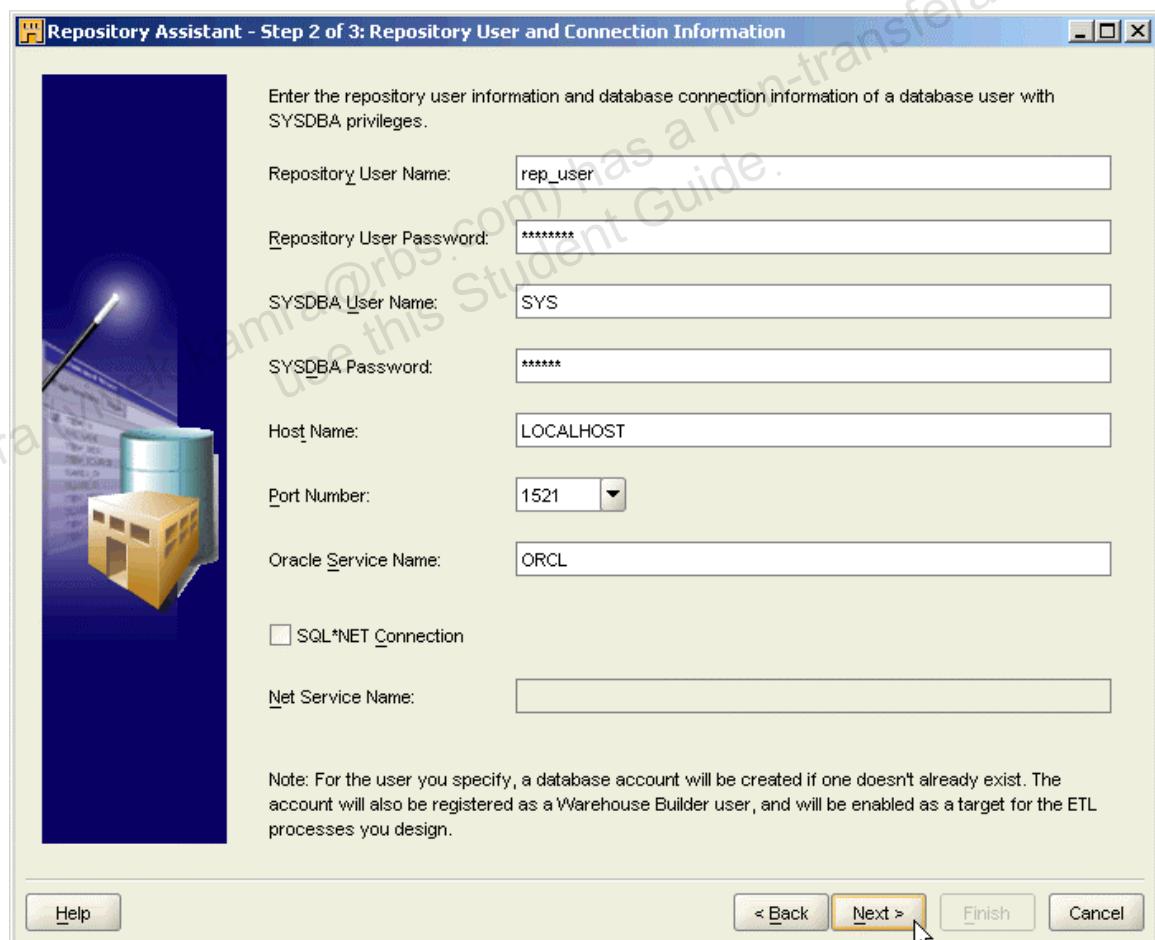
- 1) Invoke the Oracle Warehouse Builder Design Center by double-clicking the OWB10gR2 Design Center icon on the desktop.
- 2) In the Design Center Logon window, click the Get Started button. Click the Show Details button if you are not able to see the Get Started button.



The Repository Assistant Welcome page appears.

Practice 1-1: Create a Repository Owner and User (continued)

- 3) Click Next on the Repository Assistant Welcome page to launch the Install Type page.
- 4) Select Basic Install on the Install Type page, and click Next to launch the Repository User and Connection Information page.
- 5) On the Repository User and Connection Information page, create a new user **rep_user** with the following details:
 - a) Repository User Name: **rep_user**
 - b) Repository User Password: **rep_user**
 - c) SYSDBA User Name: **SYS**
 - d) SYSDBA Password: **oracle**
 - e) Host Name: **LOCALHOST**
 - f) Port Number: **1521**
 - g) Oracle Service Name: **ORCL**



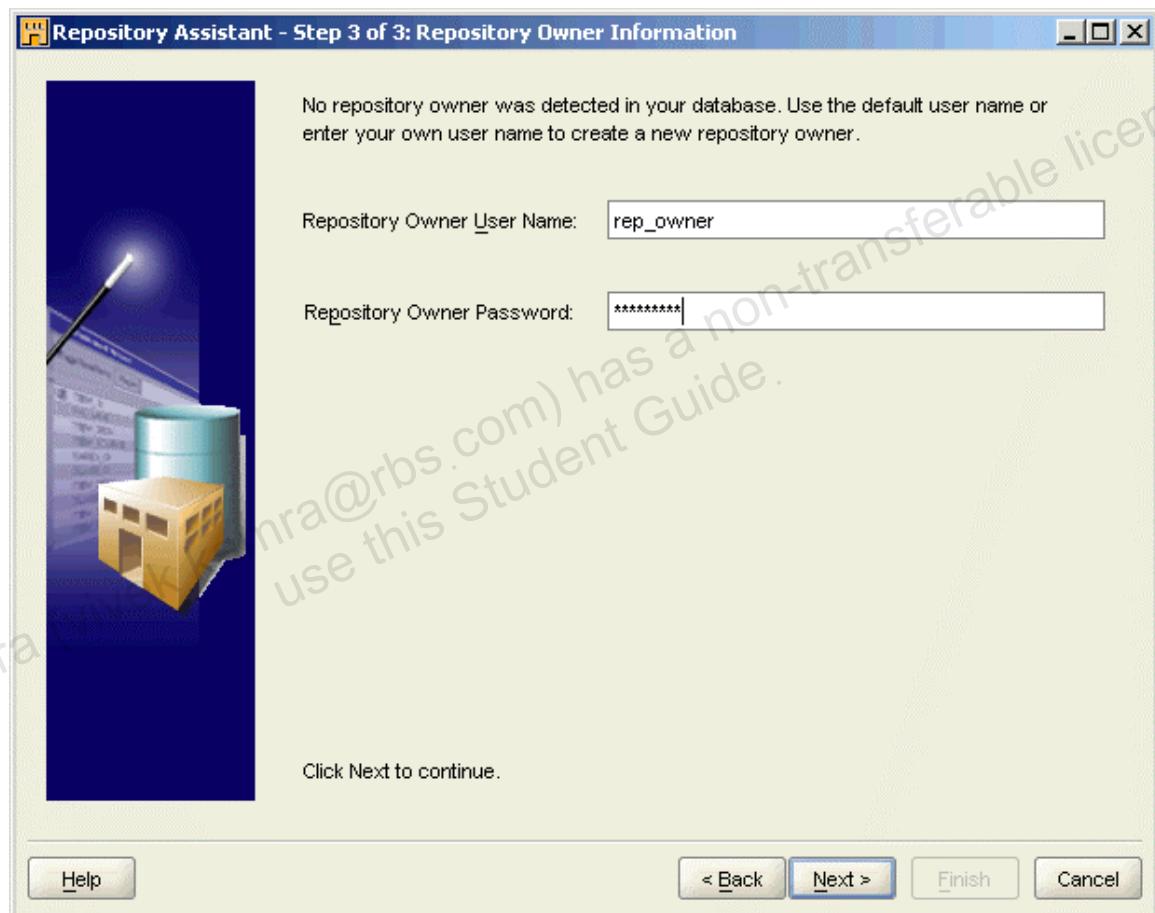
- 6) In the Password Confirmation dialog box, confirm the password, and click **OK**.
 - a) Re-enter Password: **rep_user**

Practice 1-1: Create a Repository Owner and User (continued)

- 7) On the Repository Owner Information page, create the owner named **rep_owner**. In the Password Confirmation dialog box, reenter the password.

On the Repository Owner Information page, if no owner has yet been defined, Warehouse Builder presents a default owner name (OWB_10_2_0_1_5) based on the version of Warehouse Builder installed. You instead provide a different owner name. The repository owner is a highly privileged database user with access to additional security features.

- a) Repository Owner User Name: **rep_owner** (type over the default)
- b) Repository Owner Password: **rep_owner**

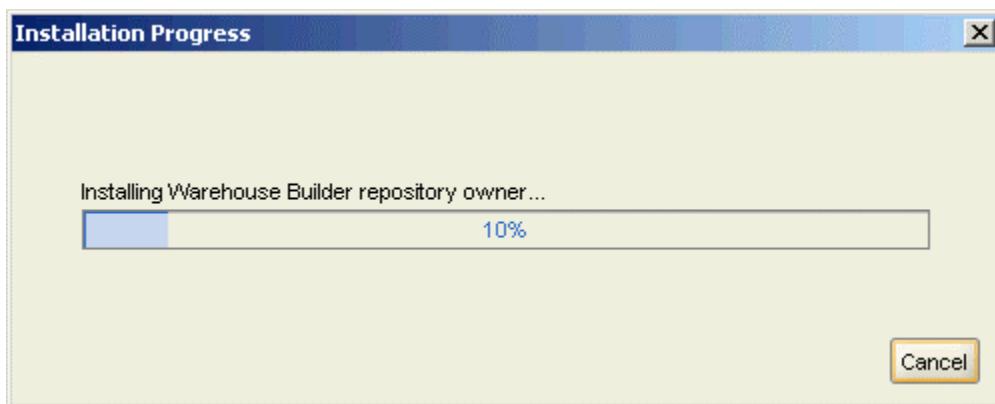


Note: If someone has previously created another repository owner on this computer, you will be unable to type over the existing name in the Repository Owner User Name field. In that case, click Cancel on the Repository Owner Information page, and create rep_owner by invoking the Repository Assistant as follows: Select Start > Programs > Oracle – OwbHome10gR2 > Warehouse Builder > Administration > Repository Assistant. Choose Advanced Setup and follow the prompts to create a new repository owner, naming it rep_owner. Then rerun steps 1 through 5. In step 6, choose rep_owner.

- 8) On the Summary page, examine the information, and click Finish.

Practice 1-1: Create a Repository Owner and User (continued)

- 9) An Installation Progress window appears. Installation of the repository owner will take a number of minutes. Your instructor may resume lecturing during this time.



- 10) An Installation Successful window appears eventually. Click OK. The Repository Assistant closes. If the Design Center Logon window remains open, click Cancel to close it. In the Warehouse Builder Warning window, click Yes to quit Warehouse Builder. You will log in later in this lesson.

Practice 1-2: Invoking OWB and Opening a Project

In this practice, you invoke the Warehouse Builder Design Center, log in as the user, open the existing project named “My Project,” create a new project, and examine the project’s properties.

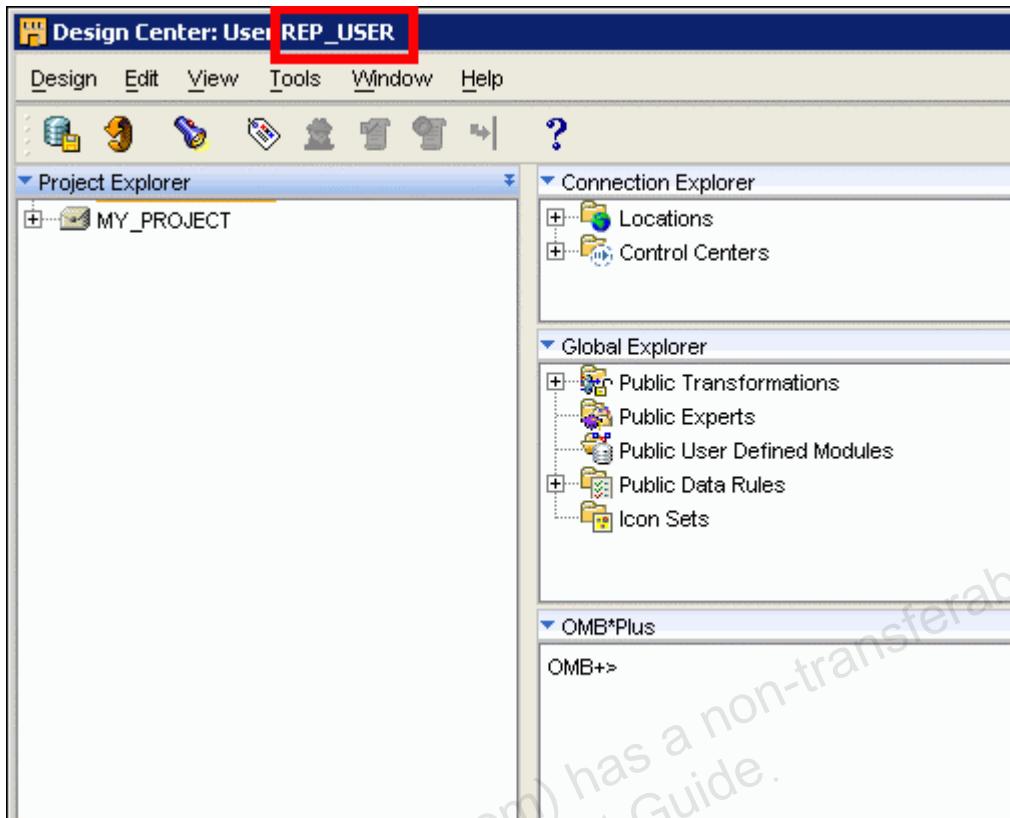
- 1) Start the OWB Design Center by using the Desktop icon or the Windows Start button.
- 2) Click **Show Details** to expand the Design Center Logon window. Enter the connection details as:
 - a) Host: **localhost**
 - b) Port: **1521**
 - c) Service Name: **orcl**

In the top portion of the Design Center Logon window, enter the username and password that you created in the previous practice:

- a) User Name: **rep_user**
- b) Password: **rep_user**



Practice 1-2: Invoking OWB and Opening a Project (continued)



Creating a New Project with the Create Project Wizard

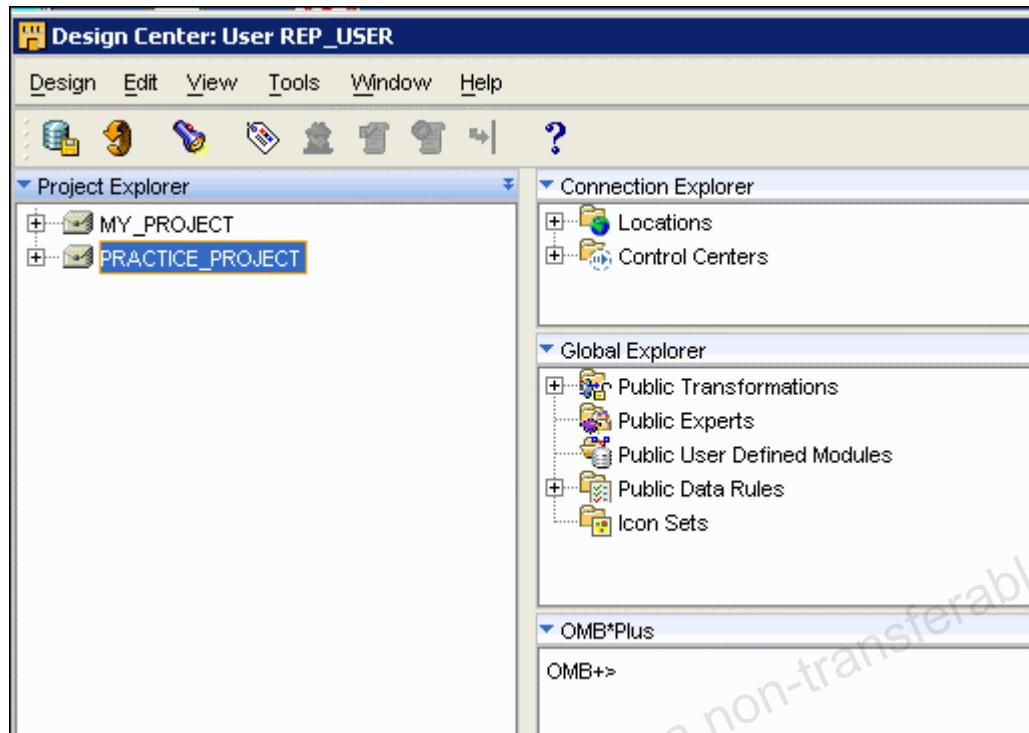
The Design Center is divided into three panels: Project Explorer, Connection Explorer, and Global Explorer. In the Project Explorer, you see MY_PROJECT, an empty default project created when Warehouse Builder is installed.

- 3) In the Design Center, select New from the Design menu. The Create Project dialog box appears. Enter the following information:
 - a) Name: Practice_Project
 - b) Description: Project for exploring the OWB client user interface

Notice that your new project is listed in the Project Explorer.

- 4) Examine all the nodes in the PRACTICE_PROJECT project tree.

Practice 1-2: Invoking OWB and Opening a Project (continued)



- 5) After examining the entire node, collapse the tree view by selecting Collapse All from the View menu.

Viewing Project Properties

- 6) Check the session properties by using the Session Properties option from the Help menu.

Viewing Project Preferences

- 7) Examine the default User Preference by selecting the Preferences option from the Tools menu. In the Preferences window, click each node on the left and examine the value on the right. **Note:** For now, do not change any settings. Later in the course, you learn more about changing these settings as and when required.

Deleting the Practice Project

- 8) Delete the project (PRACTICE_PROJECT) you just created.

Saving Your Work

- 9) Save your work using the Save All icon on the toolbar, or the Save All option from the Design menu.



- 10) After saving your work, exit OWB.

Practice for Lesson 2

In this practice, you log in to the Design Center as rep_owner and use the Metadata Import dialog box to import Warehouse Builder metadata contained in the lab2.mdl export file. The partially predefined CLASS_PROJECT is created and used in this course's remaining lessons.

Practice 2-1: Importing Metadata

You create the partially predefined project CLASS_PROJECT by using MDL Import.

Log in to the Warehouse Builder Design Center.

- 1) Log in to the Warehouse Builder Design Center with the following details:

User Name: **rep_owner**

Password: **rep_owner**

Security Settings

For the purpose of this course, you change the value for “Persist location password in metadata” security parameter. This eliminates the need for you to repeatedly provide passwords.

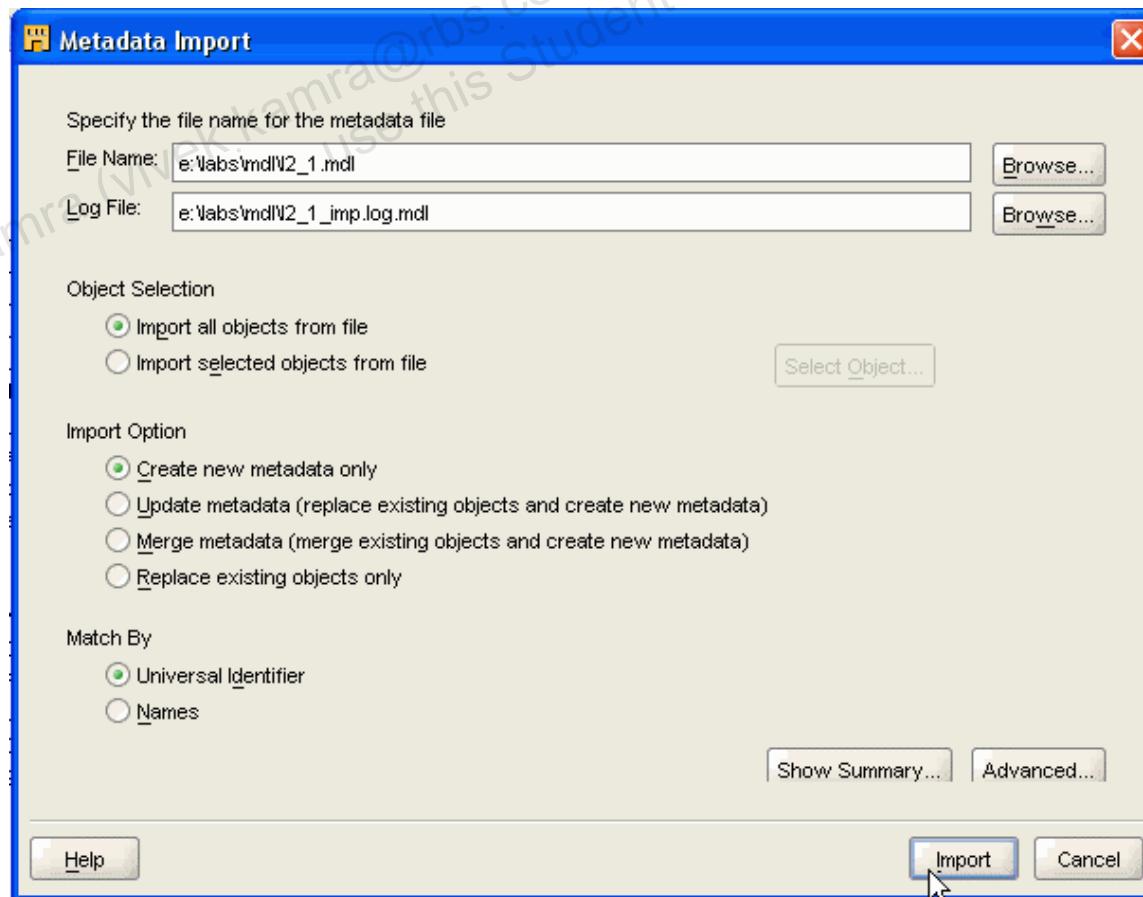
- 2) In the OMB*Plus panel, enter the following command:

```
OMBALTER USER 'REP_OWNER' SET PROPERTIES \
(DESIGNREPOS_PWD_PERSIST) VALUES ('true')
```

You now import the metadata that will create the partially predefined project.

Create the Metadata Using MDL Import

- 3) Launch the Metadata Import Utility, and import the **e:\labs\mdl\L2_1.mdl** file.

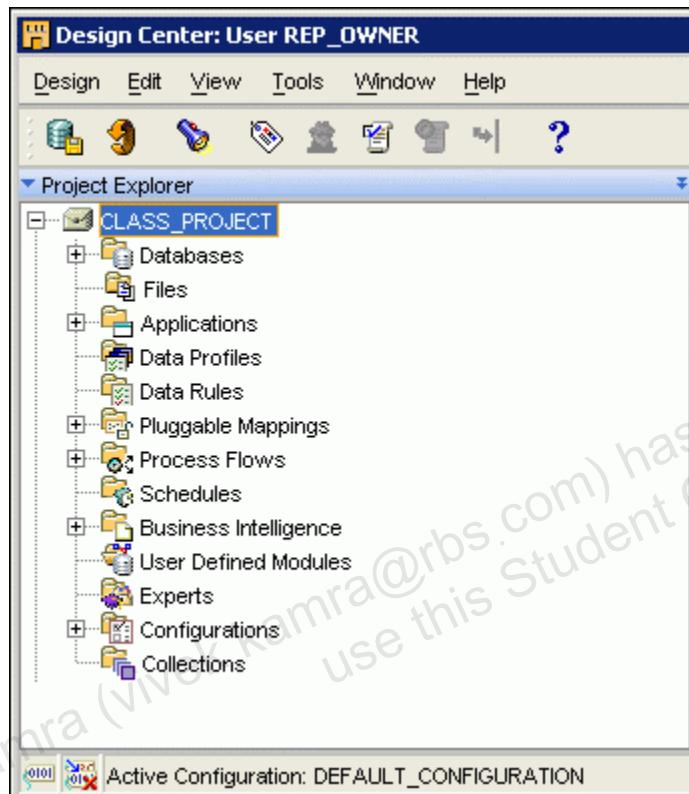


Practice 2-1: Importing Metadata (continued)

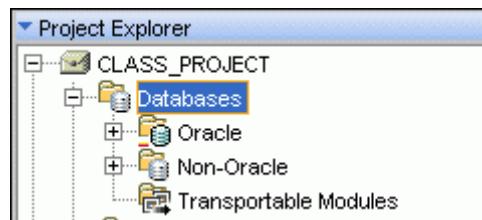
Examine the CLASS_PROJECT Metadata

A project is a container to manage your design work. Warehouse Builder contains wizards, object editors, property sheets, and object-finding tools that assist you in designing your business intelligence system.

- 4) In the Design Center, in the Project Explorer, expand CLASS_PROJECT. Various object types appear in the tree: Databases, Files, Applications, Data Profiles, and so on.

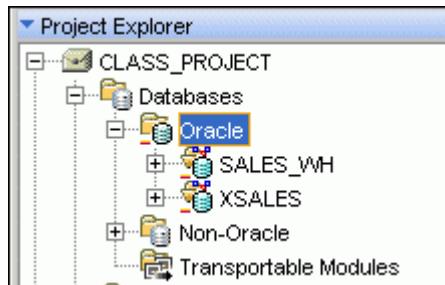


- 5) Expand the Databases node, and you see various object types that can participate in your design: Oracle, Non-Oracle, and Transportable Modules.

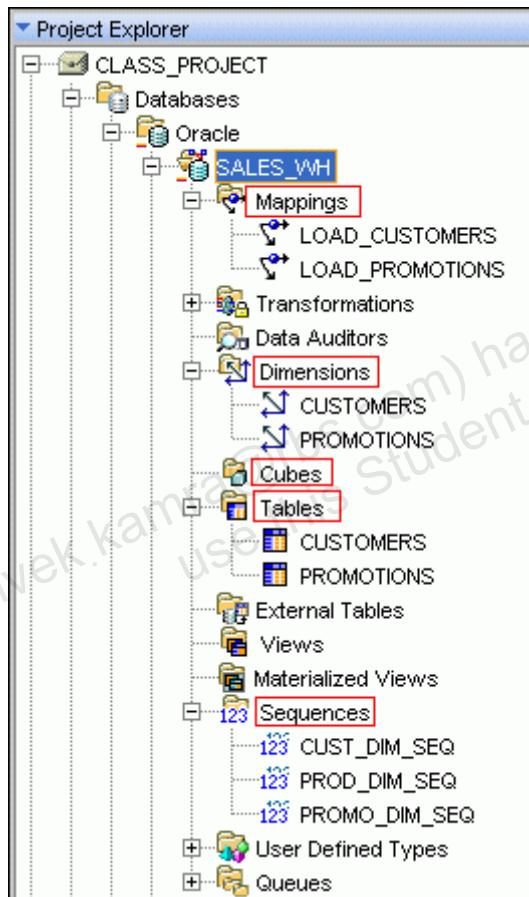


- 6) Expand the Oracle node, and you see two modules predefined by the setup scripts: SALES_WH and XSALES. Modules are logical groupings of source or target definitions.

Practice 2-1: Importing Metadata (continued)



- 7) Expand the SALES_WH module. Various object types appear in the tree: Mappings, Transformations, Dimensions, Cubes, Tables, and so on. Expand some of these nodes to see the objects that have been predefined by the setup scripts.



Practice for Lesson 3

This lesson has two parts. In part 1, you create a flat file source module. In part 2, you create an Oracle database module.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 3-1: Create a Flat File Source Module

In this practice, you create a flat file source module. The flat file module stores the definitions of the flat files. After you create the flat file module, you import metadata from a flat file by using the Import Metadata Wizard.

Create a Source Module

- 1) Create a flat file source module by using the Create Module Wizard, with the following details:

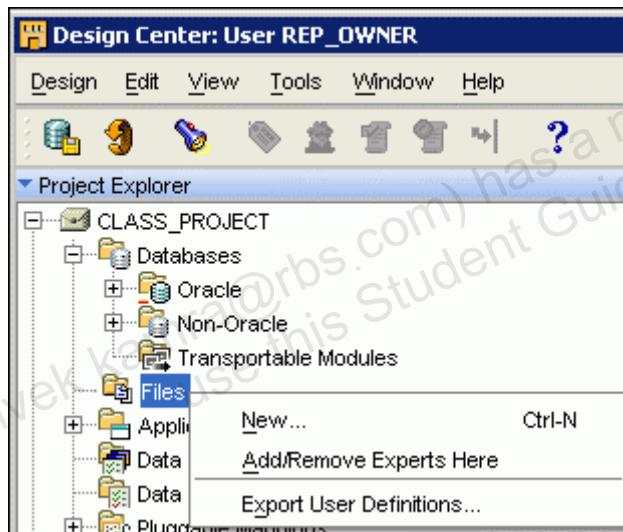
Module name: **FILE_SOURCE**

Description: **This is a Flat file source module for the Geography.txt file**

Change the location name to **FILE_SOURCE_LOCATION**.

File system location: **e:\labs\sourcefiles\GEOGRAPHY.txt**

On the Connection Information page, ensure that “Import after finish” is selected.



- 2) Import the **GEOGRAPHY.txt** flat file.
- 3) Sample the **GEOGRAPHY.txt** file by using the Flat File Sample Wizard.
- 4) Save your work.



Practice 3-2: Create a Relational Database Source Module

In this practice, you create a source module for relational database sources and import source object metadata into the source module.

Create a Relational Database Source Module

- 1) Create an Oracle source module with the following details:

Practice 3-2: Create a Relational Database Source Module (continued)

Source module name: HR

Description: This is an Oracle source module for the HR schema

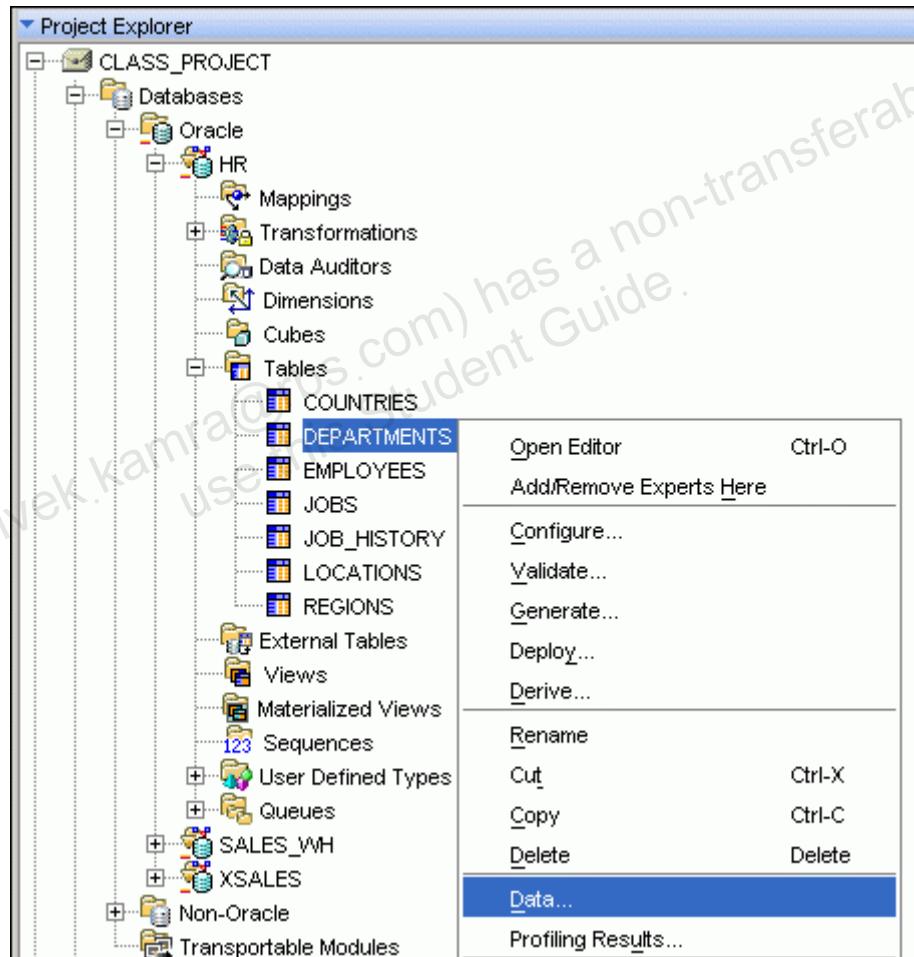
Username/Password: HR/HR

- 2) Import all the tables from the HR schema.

View Data in the HR Schema

Note: Here you are viewing data from the **HR source schema that already has data in the various tables in the schema**. OWB saves you the trouble of logging in to SQL*Plus and your having to know the syntax of SQL commands.

- 3) View the DEPARTMENTS table data.



- 4) Close the Relational Data Viewer.
- 5) Save your work.

Practice for Lesson 4

In this practice, you create a data profile, and profile the data. This practice uses a slightly modified HR schema in your local database.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 4-1: Profile the Data

In this practice, you create a profile, and profile the data.

Before you begin creating the data profile, check whether you have access to source schema that you are going to profile. In this example, it is the HR schema.

- 1) Use the Connection Explorer to examine whether you have access to the HR schema.

Create a Data Profile

- 2) Create a Data Profile with the following details:

Name: DP_HR

Objects to be included: All tables from the HR schema

- 3) Check whether the Use In Relationship Discovery property is selected for both JOBS.JOB_ID and EMPLOYEES.JOB_ID.
- 4) Profile the data.

- a) In the Data Profile Setup dialog box, enter the SYS user schema credentials as:

Password: oracle

- b) Details for the new profiling schema:

Password: **profileworkspace**

Note: You will be prompted to complete the Data Profile setup. The Data Profile setup is to create the data profiling workspace. The profiling workspace is a schema that OWB uses to store the results of the profiling job.

- 5) Save your work, and close the Data Profile Editor.

Practice 4-2: View and Analyze the Profile Results

After having created the DP_HR data profile in Practice 4-1, you are now going to view the results and analyze the profile results.

View the Profile Results

- 1) View the EMPLOYEES table data in the Profile Results Canvas.

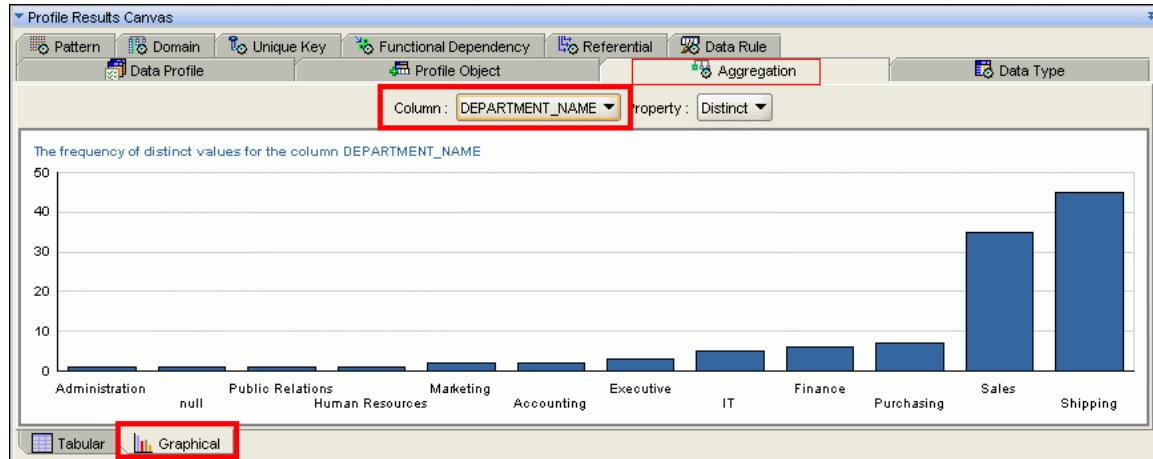
Note: Open the Data Profile Editor.

Attribute Analysis

- 2) You perform various kinds of attribute analysis with the EMPLOYEES table data.
 - a) List the maximum salary paid to an employee.
 - b) After finding the maximum salary value, find the details about the employee who earns this maximum salary.
 - c) Graphically display the distribution of employees across different departments.

Note: Use the Graphical subtab at the bottom of the Profile Results Canvas.

Practice 4-2: View and Analyze the Profile Results (continued)



- d) Find out the data type for the DEPARTMENT_NAME column.
- e) How many rows comply with the data type of the DEPARTMENT_NAME column?
- f) Are there any departments that have not been assigned any department name? Display employee details that are not assigned to any department.
- g) Find out the valid values for the JOB_ID column.
- h) Find the constraint name for the unique key constraint for the PHONE_NUMBER column.
- i) Find whether there are any phone numbers that do not comply with the unique key constraint.
- j) Display the phone number that does not comply with the unique key constraint.
- k) Find the employees who share the same number.

Now you work with the next type of Data Profiling, that is, Referential Analysis. You see existing foreign keys that were documented in the data dictionary, as well as relationships found by data profiling.

3) Perform Referential Analysis

- a) Note down the relationship name and type for the EMPLOYESS.JOB_ID column.
- b) Find out the percentage of records that comply with this foreign key relationship.
- c) Which JOB_IDS did not comply with the foreign key relationship?
- d) View the noncompliant records in a graphical view.
- e) Display the JOB_IDS found in both the EMPLOYEES and JOBS tables.
- f) Display the JOB_IDS found in the EMPLOYEES table, but not found in the JOBS table (also called orphans).
- g) Display the JOB_IDS found in the JOBS table but not found in the EMPLOYEES table (also called childless).

Practice 4-2: View and Analyze the Profile Results (continued)

Another type of data profiling is the functional dependency. The Functional Dependency tabbed page displays information about the attribute or attributes that seem to depend or determine other attributes.

Functional Dependencies Analysis

- 4) Find out which attributes are dependent, and on which attribute they depend.
- 5) Close the Data Profile Editor.
- 6) Save your work.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice for Lesson 5

In this practice, you create a target user that is your actual data warehouse schema. You then create the staging warehouse module and an external table.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 5-1: Create an External Table

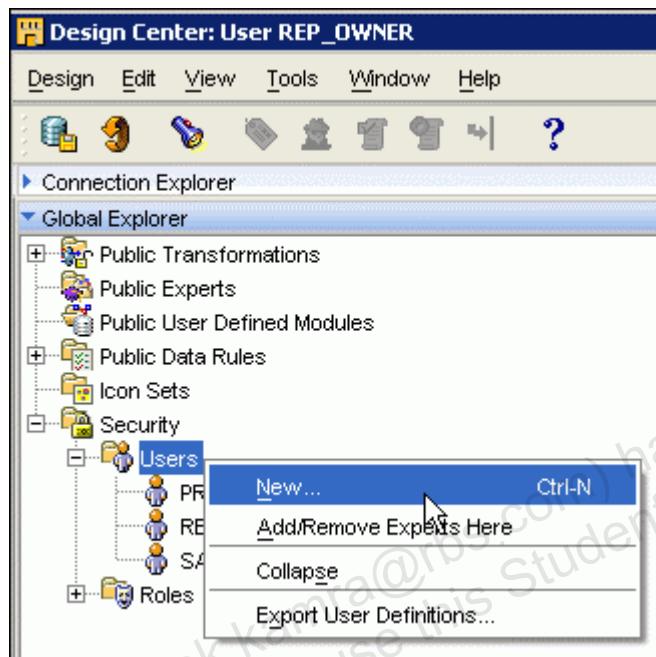
In this practice, before you create an external table, you create a target user, a target module that acts as a staging module, and finally the external table.

Create a Target User

- 1) Create a target user with the following details:

Name: **STAGING_USER**

Password: **STAGING_USER**



Create a Target Module

- 2) Create a target module named **STAGING_AREA**.
- 3) Save your work.

Create an External Table

- 4) Under the **STAGING_AREA** target module, create the **GEOG_EXT** external table with the following details:

(**Note:** The external table represents data from nonrelational sources in a relational table format.)

- a) External table name: **GEOG_EXT**
- b) File name: **FILE_SOURCE -> GEOGRAPHY.TXT**
- c) Default Location: **FILE_SOURCE_LOCATION**

Configuration Setting for the External Table

- 5) Configure the physical file system details for the external table you created.
- 6) Save your work.

Practice 5-2: Create a Simple Mapping from Source Objects to the Target

In this practice, you create your first mapping to map objects from the source to the target.

The mapping is designed to load data from the GEOG_EXT source external table to the STG_GEOG staging table.

- 1) Create a mapping named **LOAD_STG** to load data into the STG_GEOG staging table.
- 2) Add the GEOG_EXT external table.
- 3) Create an unbound table and name it STG_GEOG.

Add a Filter

- 4) Drag a filter operator named REGIONS to the canvas.

Connect Groups

- 5) Connect OUTGRP1 of the GEOG_EXT external table operator to INOUTGRP1 of the REGIONS filter operator.
- 6) Maximize the GEOG_EXT external table operator to be able to see all the attributes.
- 7) Save your work.

Define the Filter Expression

- 8) For the REGIONS filter, specify the following filter expression:

INOUTGRP1.RGN_LONG_DESC LIKE 'AMERICAS' (case sensitive)

Connect REGIONS and STG_GEOG

- 9) Drag a connection line from INOUTGRP1 of REGIONS to STG_GEOG.

To be able to see GEOG_EXT as in this screenshot, minimize the GEOG_EXT external table operator to icon size.

- 10) Minimize the GEOG_EXT external table operator.

- 11) Create the STG_GEOG table in the repository.

Validate and Generate the Mapping in the Mapping Editor

- 12) Save your work.

- 13) Validate the LOAD_STG_GEOG mapping.

- 14) Generate the LOAD_STG_GEOG mapping.

Note: When you generate a mapping, OWB implements the SQL code to perform DML and DDL commands that are necessary to move data from the sources to the mapped target.

- 15) Close the Mapping Editor

To see the results of your work, deploy the mapping and then execute the mapping.

- 16) Launch the Control Center Manager.

**Practice 5-2: Create a Simple Mapping from Source Objects to the Target
(continued)**

- 17) Register the STAGING_USER_LOCATION target location.
- 18) Deploy the LOAD_STG_GEOG mapping and the dependent objects.

You now execute the mapping that will load the data into the STG_GEOG table.

- 19) Execute the LOAD_STG_GEOG mapping.
- 20) Close the Control Center Manager.
- 21) View the data in the STG_GEOG table.
- 22) Close the Relational Data Viewer.
- 23) Save your work.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice for Lesson 6

In this practice, you learn how to derive data rules and run correction mappings.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 6-1: Deriving Data Rules and Creating Correction Mappings

Now that you are familiar with creating a target module, location, and target user, you run this script that will create the target user DP_TGT, a target module DP_TGT, and a location DP_TGT_LOCATION for you. Also, the DP_TGT target user is granted privileges to access tables from the HR schema.

- 1) Run the `create_dp_tgt.tcl` file in the OMB*Plus window in the Design Center.
- 2) Open the DP_HR data profile.
- 3) Display the Domain values for JOB_ID in the EMPLOYEES table.
- 4) Derive a data rule for this detected domain.
- 5) Generate a correction mapping for the JOB_ID rule that you created.
- 6) Save your work.
- 7) Close the Data Profile Editor.

Deploy and Run Correction Mappings

- 8) Launch the Control Center Manager.
- 9) Register DP_TGT_LOCATION.
- 10) Deploy the mapping and its related objects.

Execute the Mapping

- 11) Execute the M_EMPLOYEES mapping.
- 12) In the Job Details window, examine the results of execution.
- 13) Close the Job Details window and the Control Center Manager.
- 14) Check the result of your work. There should be no entries such as PU_CLIRK or SA_REPP.
- 15) Close the Relational Data Viewer.

Practice for Lesson 7

In this practice, you complete designing the partially created CLASS_PROJECT data warehouse project.

In Part 1 of the practice, you create the CHANNELS dimension using the wizard.

In Part 2 of the practice, you create the PRODUCTS dimension using the editor.

In Part 3 of the practice, you create the TIMES dimension using Time wizard.

In Part 4 of the practice, you create the SALES cube using the editor.

In Part 5 of the practice, you design the mapping to load the PRODUCTS dimension.

OWB automatically creates for you the mapping to load the TIMES dimension. To save time during this class, you run a script that creates the LOAD_SALES mapping to load the SALES cube. The guided instructions to create the LOAD_SALES cube mapping are in the appendix titled “Create the LOAD_SALES Mapping.”

Practice 7-1: Create a Dimension Using the Wizard

In this practice, you create the CHANNELS dimension by using the Create Dimension Wizard.

- 1) Use the Connection Explorer to examine whether you have access to the XSALES source schema.

Create the CHANNELS Dimension Using the Wizard

- 2) Using the wizard, create the CHANNELS dimension under the SALES_WH target module. Make note of the following details:
 - a) Name: CHANNELS
 - b) Storage Type: ROLAP: (Relational Storage)
 - c) For Dimension Attributes follow the details as shown below:

Name	Identifier	Data Type	Length	Descriptor
ID	Surrogate	NUMBER		
NAME		VARCHAR2	60	Short Description
SOURCE_ID	Business	VARCHAR2	40	

- d) Levels page: Create three levels as:

Name	Description
TOTAL	Total
CLASS	Class
CHANNEL	Channel

- e) Level attributes page: Specify the attributes for each level as follows:

Level	Level Attributes
TOTAL	ID
	NAME
	SOURCE_ID
CLASS	ID
	NAME
	SOURCE_ID

Practice 7-1: Create a Dimension Using the Wizard (continued)

CHANNEL	ID
	NAME
	SOURCE_ID

The CHANNELS dimension is created for you. You now configure one of the dimension properties.

- 3) Assign **Deploy All** as the deployment option for the CHANNELS dimension.
- 4) Save your work.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 7-2: Create a Dimension Using the Editor

In this practice, you create the PRODUCTS dimension by using the editor.

- 1) Create the PRODUCTS dimension by using the editor. The dimension should be created under the SALES_WH target module and with the following details on the various tabs:

- a) Name tab:

Name: PRODUCTS

Description: Products Dimension

- b) Storage tab:

Accept default option ROLAP: (Relational Storage)

Implementation: Star

- c) Attributes tab:

Sequence: Select **PROD_DIM_SEQ** from the list.

Name	Identifier	Data Type	Length
ID	Surrogate	Number	
Name		Varchar2	60
Description		Varchar2	100
Source_Id	Business	Varchar2	40
Pack_Size		Varchar2	30
List_Price		Varchar2	10

- d) Levels tab:

Level	Description
TOTAL	Total
CATEGORY	Category
SUBCATEGORY	Sub Category
PRODUCT	Product

Select the following attributes for each level:

Level	Level Attributes

Practice 7-2: Create a Dimension Using the Editor (continued)

TOTAL	ID
	Name
	Description
	Source_Id
CATEGORY	ID
	Name
	Description
	Source_Id
SUBCATEGORY	ID
	Name
	Description
	Source_Id
PRODUCT	ID
	Name
	Description
	Source_Id
	Pack_Size
	List_Price

e) Hierarchies tab:

Hierarchy: **PROD_STD**

Levels for PROD_STD:

Level
TOTAL
CATEGORY

Practice 7-2: Create a Dimension Using the Editor (continued)

SUBCATEGORY
PRODUCT

f) SCD tab:

Accept the default selection, Type 1: Do not keep history.

2) Create the dimension table, and bind it to the repository. (Use Auto Bind.)

You have now created the PRODUCTS dimension.

3) Select **Deploy All** as Deployment Options for the PRODUCTS dimension.

4) Save your work, and close the Data Object Editor.

5) Set the business name of the PRODUCTS dimension.

Practice 7-3: Create a TIMES Dimension Using the Time Wizard

In this practice, you create the TIMES dimension by using the Time Wizard.

- 1) Create the TIMES dimension by using the Time Wizard. Create this TIMES dimension under the SALES_WH module with the following details:

- a) Name and Description page:

Name: TIMES

Description: Times Dimension

- b) Storage Type page:

Accept the default, ROLAP: Relational Storage.

- c) Data Generation page:

Start year: 2001

Number of years: 1

- d) Levels page:

Normal Hierarchy

Levels in the Normal Hierarchy

Calendar Year

Calendar Quarter

Calendar Month

- 2) Select **Deploy All** as Deployment Options for the TIMES dimension.

- 3) Save your work.

Practice 7-4: Create a Cube Using the Editor

In this practice, you create a cube by using the editor.

Create the SALES Cube

- 1) Using the editor, create the sales cube with the following details:

- a) Name tab:

Name: SALES

- b) Storage tab:

ROLAP: Relational Storage

Select Create Bitmap Indexes.

- c) Dimensions tab:

DIMENSION	LEVEL
TIMES	CALENDAR_MONTH
PRODUCTS	PRODUCT
CHANNELS	CHANNEL
CUSTOMERS	CITY
PROMOTIONS	SUBCATEGORY

- d) Measures tab:

Name	Description	Data Type	Precision	Scale
Amount	Sales Amount	Number	10	2
Quantity	Sales Quantity	Number		
Cost	Sales Cost	Number	10	2

- e) Aggregation tab

After defining the cube structure, you now specify the details about the database tables or views that store the cube data.

- 2) Create and bind the table to the repository.
- 3) Configure Deployment Options for the SALES cube to **Deploy All**.
- 4) Save your work, and close the Data Object Editor.
- 5) Set Business Name for the SALES cube to SALES.

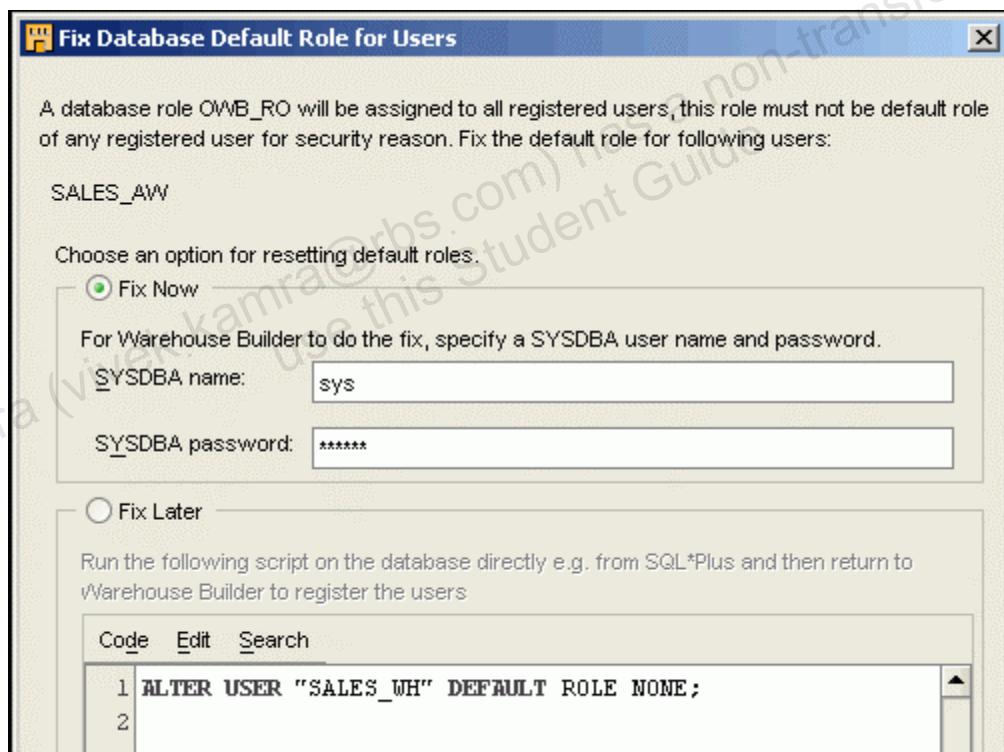
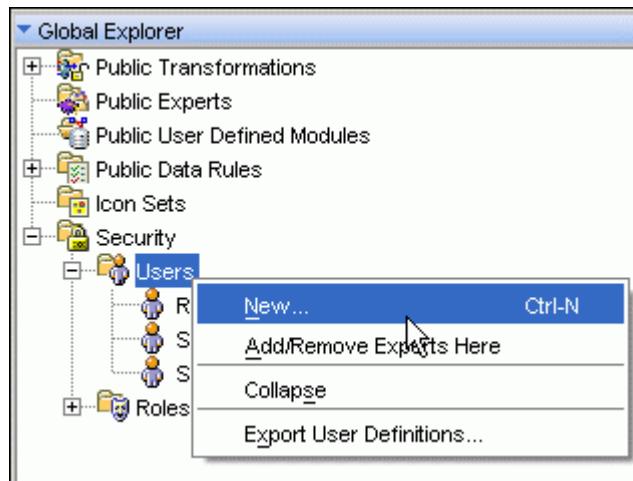
Practice 7-5: Design ETL Mappings

In this practice, you design a mapping to load a dimension. Before you start creating a mapping, you register the SALES_WH database user as the target user for this class.

Practice 7-5: Design ETL Mappings (continued)

Registering a Database User as a Target User

- 1) Register the SALES_WH user as a target user. Use the Security node in the Global Explorer panel.



The Register Users Progress dialog box displays the progress of the user creation. The target user definition is complete.

Note: For security reasons, OWB does not allow you to register database users with default roles in the database set to ALL. Warehouse Builder gives you options for changing the default setting. You correct the role by selecting Fix Now. Selecting the Fix Now option registers the SALES_AW user and assigns a database role OWB_<repository name> to the user.

- 2) Save your work.

Practice 7-5: Design ETL Mappings (continued)

- 3) Use the Connection Explorer to examine whether you have access to the SALES_WH schema.

Design the LOAD_PRODUCTS Mapping to Load the PRODUCT Dimension

- 4) Create a mapping named LOAD_PRODUCTS.
- 5) Drag the following tables from the XSALES module to the canvas:
 - a) XSALES.CATEGORIES
 - b) XSALES.PRODUCTS
- 6) Change the PRODUCTS table to PRODUCTS_IN.
- 7) Drag the PRODUCTS dimension to the canvas from the SALES_WH module.
- 8) Change the PRODUCTS dimension to PRODUCTS_OUT.
- 9) From the Palette, drag the following three operators to the canvas:
 - a) CONSTANT
 - b) FILTER
 - c) FILTER
- 10) Rename the CONSTANT operator to TOTALS.
- 11) To the TOTALS constant operator, add the following two attributes with the following details:

Name	Data Type	Length
TOT	VARCHAR2	20
SRC_ID	VARCHAR2	20

- 12) Add the following expression to the two attributes of the TOTAL constant operator:

Attribute	Expression
TOT	'Product Total'
SRC_ID	'99999'

- 13) Change the FILTER operator to CATS.
- 14) Add connection lines from attributes in the CATEGORIES.INOUTGRP1 operator to INOUTGRP1 of the CATS operator as:

Practice 7-5: Design ETL Mappings (continued)

CATEGORIES.INOUTGRP1	CATS
ID	INOUTGRP1
NAME	INOUTGRP1
DESCRIPTION	INOUTGRP1
CATEGORY_ID	INOUTGRP1

15) Specify the filter condition for the CATS filter operator as:

INOUTGRP1. CATEGORY_ID IS NULL

16) Change the FILTER operator to SUBCATS.

17) Add connection lines from attributes in the CATEGORIES.INOUTGRP1 operator to INOUTGRP1 of the SUBCATS operator as:

CATEGORIES.INOUTGRP1	SUBCATS
ID	INOUTGRP1
NAME	INOUTGRP1
DESCRIPTION	INOUTGRP1
CATEGORY_ID	INOUTGRP1

18) Specify the filter condition for the SUBCATS filter operator as:

INOUTGRP1. CATEGORY_ID IS NOT NULL

19) Draw connection lines from attributes in OUTGRP1 of the TOTALS constant operator to attributes in groups in the PRODUCTS_OUT dimension operator as:

TOTALS.OUTGRP1	PRODUCTS_OUT
TOT	TOTAL.NAME
TOT	TOTAL.DESCRIPTION
SRC_ID	TOTAL.SOURCE_ID
SRC_ID	CATEGORY.TOTAL_SOURCE_ID

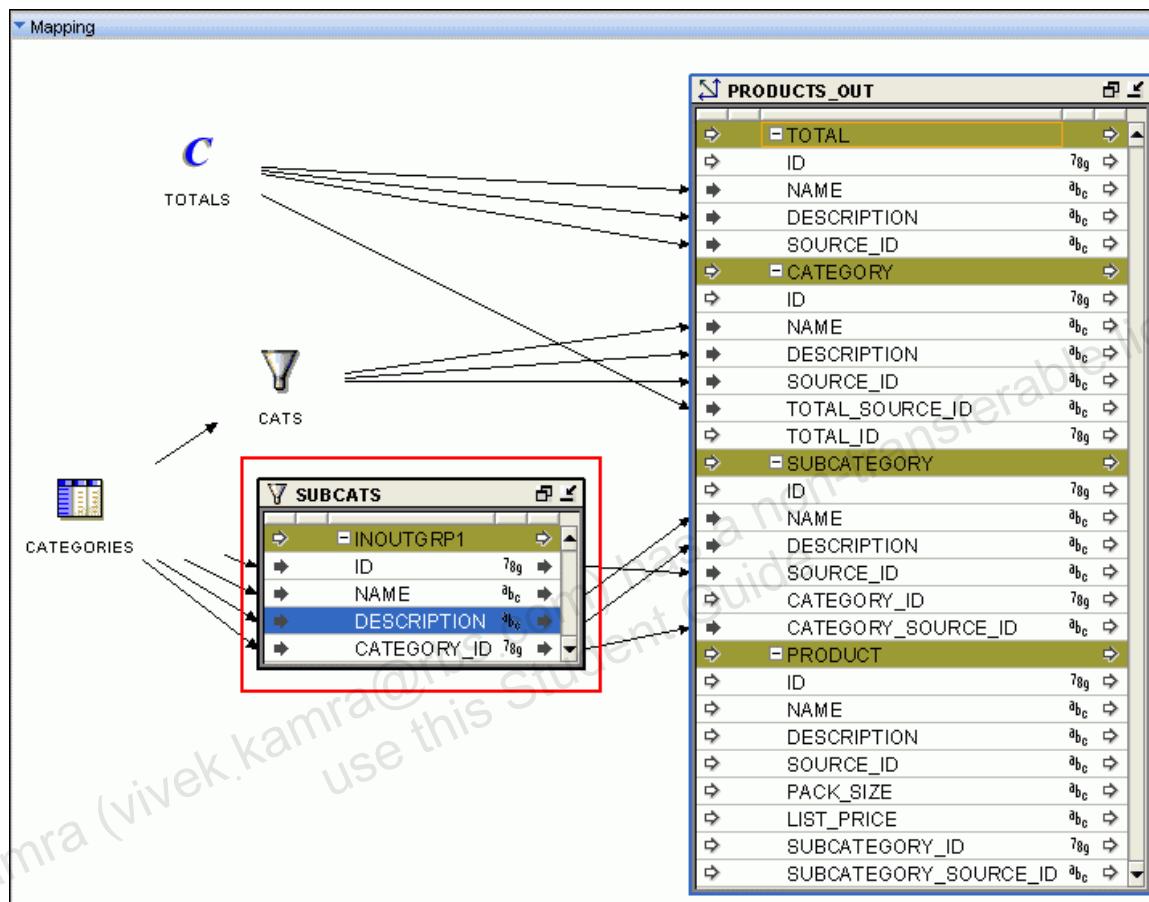
20) Draw connection lines from attributes in INOUTGRP1 of the CATS filter operator to attributes in the CATEGORY group of the PRODUCTS_OUT dimension operator.

CATS.INOUTGRP1	PRODUCTS_OUT
ID	CATEGORY.SOURCE_ID
NAME	CATEGORY.NAME
DESCRIPTION	CATEGORY.DESCRIPTION

21) Draw connection lines from attributes in INOUTGRP1 of the SUBCATS filter operator to attributes in the SUBCATEGORY group of the PRODUCTS_OUT dimension operator.

Practice 7-5: Design ETL Mappings (continued)

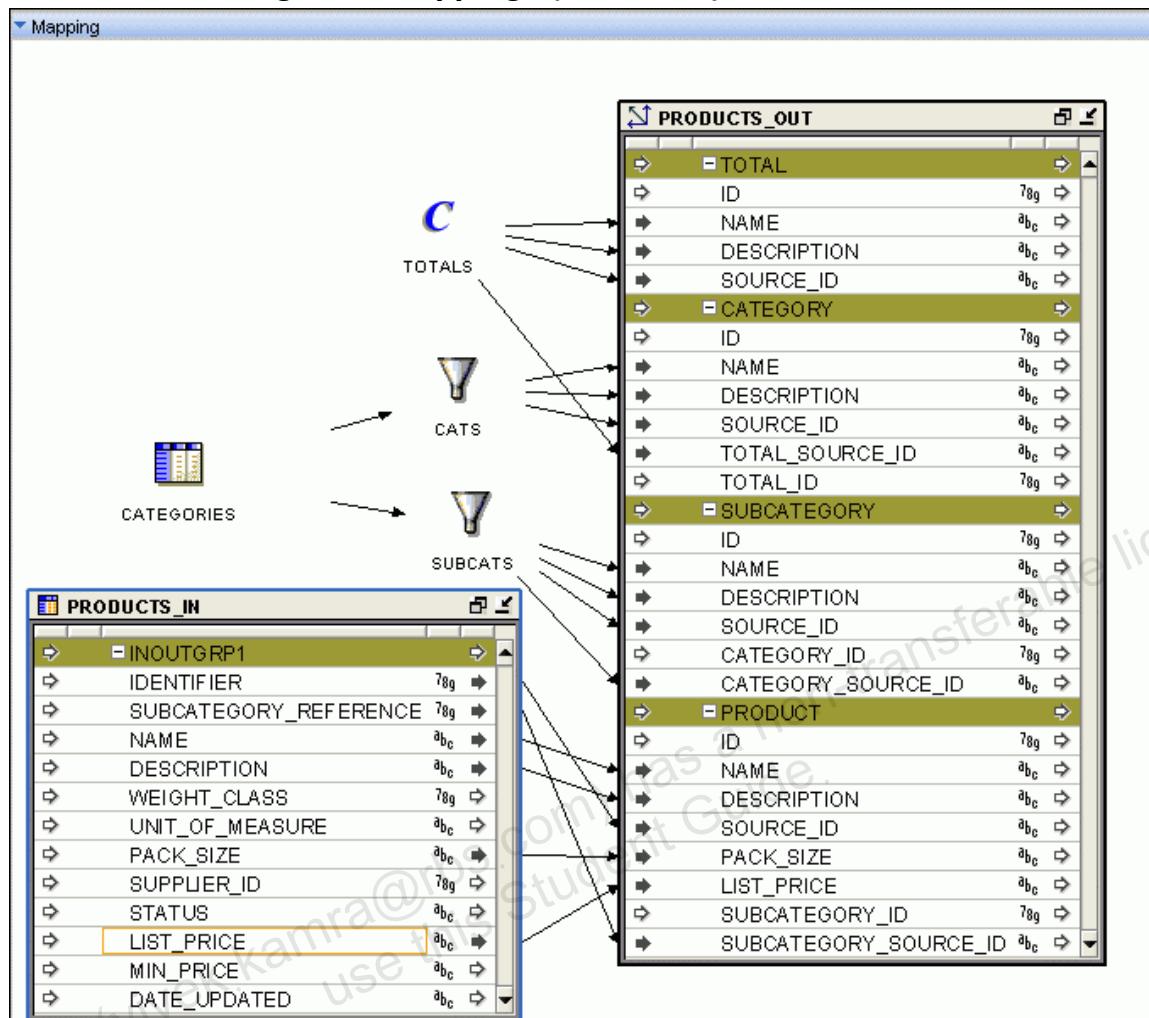
SUBCATS.INOUTGRP1	PRODUCTS_OUT
CATEGORY_ID	SUBCATEGORY.CATEGORY_SOURCE_ID
ID	SUBCATEGORY.SOURCE_ID
NAME	SUBCATEGORY.NAME
DESCRIPTION	SUBCATEGORY.DESCRIPTION



- 22) Draw connection lines from attributes in INOUTGRP1 of the PRODUCTS_IN table operator to attributes in the PRODUCT group of the PRODUCTS_OUT dimension operator.

PRODUCTS_IN.OUTGRP1	PRODUCTS_OUT
SUBCATEGORY_REFERENCE	PRODUCT.SUBCATEGORY_SOURCE_ID
IDENTIFIER	PRODUCT.SOURCE_ID
NAME	PRODUCT.NAME
DESCRIPTION	PRODUCT.DESCRIPTION
PACK_SIZE	PRODUCT.PACK_SIZE
LIST_PRICE	PRODUCT.LIST_PRICE

Practice 7-5: Design ETL Mappings (continued)



Validate the Mapping

23) Validate the LOAD_PRODUCTS mapping.

Generate the Mapping

24) Generate the LOAD_PRODUCTS mapping.

25) Save your work.

26) Close the Mapping Editor.

You completed creating the mapping to load the PRODUCTS dimension.

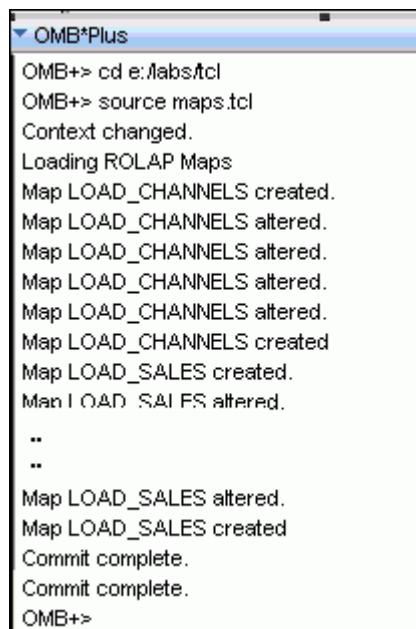
Complete the Partially Created CLASS_PROJECT

You completed designing the CHANNELS dimension, the PRODUCTS dimension, and the TIMES dimension. You then created the SALES cube and finally created the LOAD_PRODUCTS mapping to load the PRODUCTS dimension. The other objects were created for you at the initial setup. To save time during this class, the instructions to design the mapping that loads the cube is precreated for you. For your reference, the mapping to load the SALES cube is found in the appendix titled "MOLAP: Modeling and Loading."

Practice 7-5: Design ETL Mappings (continued)

Run the tcl Scripts to Complete the Design of CLASS_PROJECT

- 27) In the OMB Plus panel in the Design Center, execute the maps.tcl script to complete the design of this project.



```
OMB*Plus
OMB+> cd e:\labs\tcl
OMB+> source maps.tcl
Context changed.
Loading ROLAP Maps
Map LOAD_CHANNELS created.
Map LOAD_CHANNELS altered.
Map LOAD_CHANNELS altered.
Map LOAD_CHANNELS altered.
Map LOAD_CHANNELS altered.
Map LOAD_CHANNELS created
Map LOAD_SALES created.
Map LOAD_SALES altered.
"
"
Map LOAD_SALES altered.
Map LOAD_SALES created
Commit complete.
Commit complete.
OMB+>
```

- 28) Save your work.

Now you have all the mappings needed to complete the design of the warehouse. You use the Generate option to generate the code to implement the design.

Generate the Remaining Mappings

You designed and generated the LOAD_PRODUCTS mappings.

- 29) Generate the following mappings.

LOAD_PROMOTIONS
LOAD_CHANNELS
LOAD_CUSTOMERS
TIMES_MAP
LOAD_SALES

Practice for Lesson 8

In this practice, you learn how to create a Type 2 Slowly Changing Dimension.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 8-1: Handling Slowly Changing Dimensions

In this practice, you use the existing CHANNELS dimension and edit the SCD type from type 1 to type 2.

- 1) Open the CHANNELS dimension in the Data Object Editor.
- 2) Add the following two attributes:

Name	Data type
Effective_Date	DATE
Expiration_Date	DATE

The screenshot shows the 'Dimension Details' window for the 'SALES_WH.CHANNELS' dimension. The 'Attributes' tab is selected. A message at the top says 'Choose the sequence that will populate the Dimension and Surrogate Keys:' followed by a dropdown menu containing 'CHANNELS_SEQ'. Below this is a table listing dimension attributes:

	Name	Description	Identifier	Data Type	Length
1	ID		Surrogate	NUMBER	
2	NAME			VARCHAR2	60
3	SOURCE_ID		Business	VARCHAR2	40
4	EFFECTIVE_DATE			DATE	
5	EXPIRATION_DATE			DATE	

- 3) On the Levels tabbed page, apply the two new attributes EFFECTIVE_DATE and EXPIRATION_DATE to the CHANNEL level.
- 4) On the SCD tabbed page, select “Type 2: Store the complete change history.”
Note: The Settings button is enabled.
- 5) Click the **Settings** button. In the “Type 2 slowly changing dimension” window, select the attributes that trigger history, and set the values for the two data attributes as:

Levels	Identifying Attribute	Record History
CHANNEL	EFFECTIVE_DATE	Effective date
	EXPIRATION_DATE	Expiration date
	NAME	Trigger History

Binding the Dimension Channels to the Database Table

- 6) Propagate the changes made to the dimension object to the database table.
- 7) Save your work, and close the Data Object Editor.

Synchronize the CHANNELS Repository Object with the CHANNELS Operator

- 8) In the Mapping Editor, the CHANNELS dimension operator needs to be updated with the new CHANNELS repository object.

Practice 8-1: Handling Slowly Changing Dimensions (continued)

- 9) Save your work.
- 10) Close the Mapping Editor

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice for Lesson 9

On the basis of the business intelligence needs, you can specify how you want to deploy your design (for example, deploy to Relational Databases, Oracle Workflow, or Oracle Discoverer). In the previous lessons, you specified to deploy to an Oracle location (SALES_WH_LOCATION).

For this class, when working with process flows, you use Oracle Workflow location. Before you start designing the process flow, you create an Oracle Workflow location.

You define an Oracle Workflow location to specify where you want to deploy your business definitions. The Oracle Workflow location will point to a workflow engine (OWF_MGR) that runs in the target database. Warehouse Builder process flows comply with the XML Process Definition Language (XPDL).

Note: You must run the Workflow Configuration Assistant to load Oracle Workflow into your database.

For this classroom setup, Oracle Workflow has been loaded into your current working database. OWF_MGR is the username of your Oracle Workflow database account. The default Workflow account for a fresh installation is OWF_MGR.

The next step that you have to perform is to create an Oracle Workflow location.

Practice 9-1: Create the LOAD_SALES_WH Process Flow

Create an Oracle Workflow Location

- 1) Create an Oracle Workflow location named OWF_MGR_LOCATION with the following details:

Password: **owf_mgr**

Host: **localhost**

Port: **1521**

Service: **orcl**

Schema: **owf_mgr**

Version: **2.6.4**

- 2) Verify whether this newly created OWF_MGR_LOCATION exists.

You have completed creating the Oracle Workflow location. You now design the process flow.

Grant privileges to the owf_mgr schema

- 3) Grant the OWB_O_REP_OWNER privileges to **owf_mgr**.

Design a Process Flow

- 4) Create a process flow with the following details:

- a) Name: **PF_SALES_WH**
- b) Description: **Process Flow module for SALES_WH**
- c) Location: **OWF_MGR_LOCATION**
- d) Process Flow Package: **PK_SALES**
- e) Process Flow: **LOAD_SALES_WH**

Add a Fork Activity

- 5) In the Process Editor, drag a Fork activity to the canvas. Place it between the START activity and the END activity.

Add the Mappings

- 6) Drag the following six mappings to the canvas:

- LOAD_CHANNELS
- LOAD_PROMOTIONS
- LOAD_CUSTOMERS
- LOAD_PRODUCTS
- TIMES_MAP
- LOAD_SALES

Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)

- 7) Arrange the activities on the process flow canvas such that the following mapping activities appear after the FORK activity and before the END activity:

Add an AND Activity

- 8) Drag an AND activity to the canvas.

Add Control Activities

- 9) Drag the following activities to the canvas:

- End with Errors
- End with Warnings

Add Transitions

- 10) Connect the activities using transitions as follows:

FROM	TO
START	FORK
FORK	LOAD_CHANNELS
FORK	LOAD_PROMOTIONS
FORK	LOAD_CUSTOMERS
FORK	LOAD_PRODUCTS
FORK	TIMES_MAP
LOAD_CHANNELS	AND
LOAD_PROMOTIONS	AND
LOAD_CUSTOMERS	AND
LOAD_PRODUCTS	AND
TIMES_MAP	AND
AND	LOAD_SALES
LOAD_SALES	END_SUCCESS
LOAD_SALES	END_ERROR
LOAD_SALES	END_WARNING

- 11) Save your work.

ERROR Handling Logic: Defining Conditions for the Transitions

- 12) Specify the SUCCESS enumerated condition for the SUCCESS transition.

- 13) Specify the ERROR enumerated condition for the ERROR activity.

- 14) Specify the WARNING enumerated condition for the WARNING activity.

- 15) Save your work.

- 16) Validate the process flow.

- 17) Generate the process flow.

- 18) Close the Process Editor.

Practice 9-2: Create a Schedule

In this practice, you create a schedule that becomes active every two hours.

- 1) Create a Schedule module named SCHED_SALES_WH; the associated location is SALES_WH_LOCATION.
- 2) Create a schedule named EVERY_2_HOURS with the following details:

Time zone for this schedule drop-down select America/Los Angeles

- Date: <enter a date when you want this schedule to start>
- Time: <enter start time> 00:00:00 AM

Choose an end time for this schedule

- Date: <enter a future date>
- Time: 00:00:00 AM

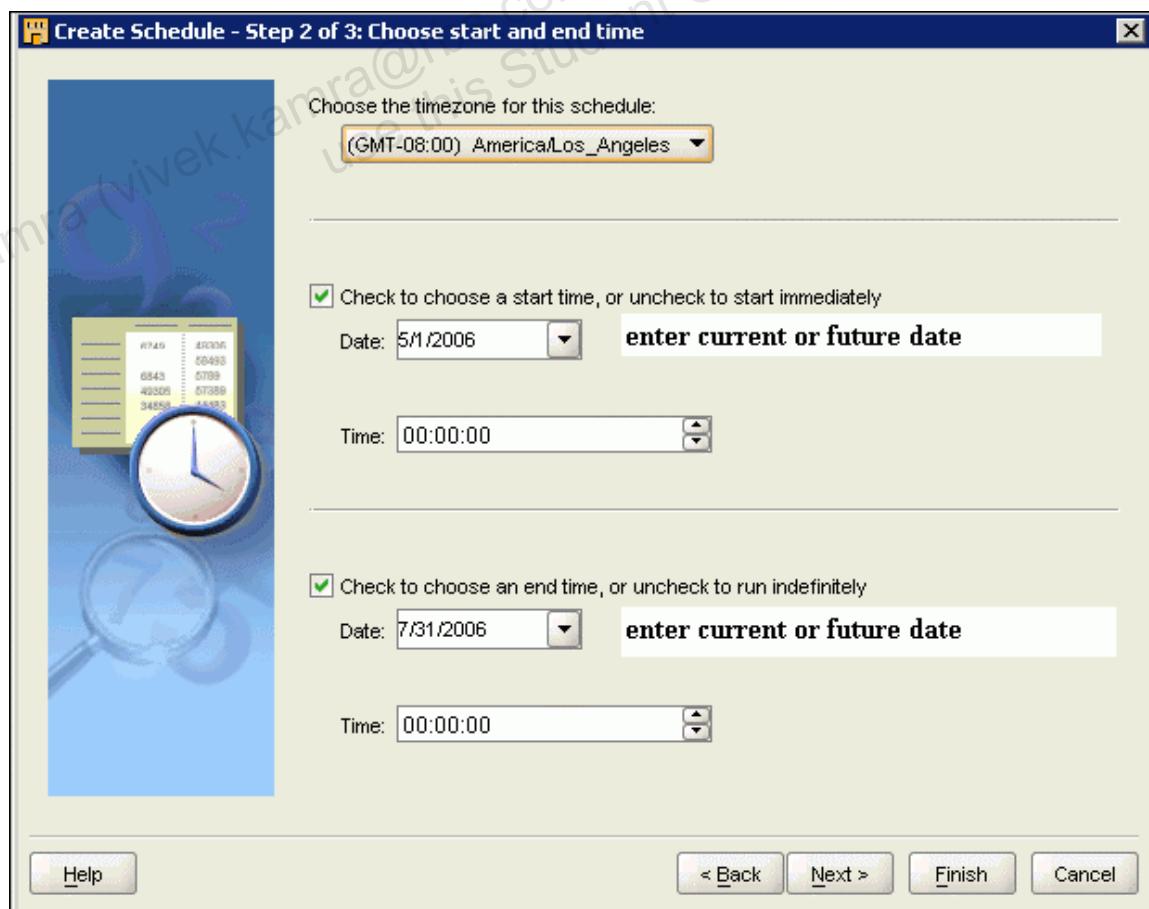
On the “Choose the frequency and repeat interval page,” specify the following:

Frequency: Daily

Repeat every: 1 days

Click **Next**. The “Choose start and end time” page is displayed.

- 3) On the “Choose start and end time” page, enter the information as shown in the screenshot:



Click **Next**. The “Choose the frequency and repeat interval” page appears.

- 4) On the “Choose the frequency and repeat interval” page, specify the following:

Frequency: **Daily**

Repeat every: **1 days**

Click **Next**.

- 5) Review the Summary page, and click **Finish**.

Editing a Schedule

The wizard does not give you all the capabilities to edit the schedule. You get more advanced scheduling capabilities if you edit the schedule.

- 6) Edit the schedule to be able to run at 9, 11, 13, 15, and 17 (5 PM).
- 7) Specify the name of the EVERY_2_HOURS schedule in the Referred Calendar property of the process flow.
- 8) Examine the scheduled job entry in the Control Center Manager.

You set up an e-mail activity to be used on a very frequent basis. You create a template that has the values preset; for example, the SMTP server, from address, and to address.

- 9) Create a template named WARNING_EMAILS for an e-mail activity.

Name	Default Value
SMTP Server	rgmareasmtplib.company.com
FROM ADDRESS	bi@company.com
TO ADDRESS	admin@company.com
IMPORTANCE	High
SUBJECT	WARNING

Create a NOTIFICATION Activity

The notification activity enables integration with Oracle Workflow’s notification mechanism. For example, users can be sent messages in the form of e-mails to which they have to respond. The response will determine the outcome of the notification.

- 10) Create a notification named PF_NOTIFICATION under a new process flow package named PK_ACTIVITIES, and enter the following values for the corresponding parameters:
- 11) Drag an End with Errors activity to the canvas.
- 12) Draw a transition and specify the condition as specified in this table:

From	To	Condition
START	NOTIFICATION	
NOTIFICATION	END_ERROR	NOT_OK
NOTIFICATION	END_SUCCESS	OK

- 13) Save your work.
- 14) Close the Process Editor.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice for Lesson 10

This practice consists of two parts. In Part 1, you deploy the relational and dimensional objects, the ETL mappings, and the process flow. You then execute the process flow, and the data is loaded into the dimensional objects. You use the Data Viewer to view the data that is loaded into the objects.

In Part 2 of this practice, you learn how to manage your metadata using the Dependency Manager in Warehouse Builder. You also use the repository browser, which is the central reporting tool for viewing the data and metadata stored in the Warehouse Builder repository.

Practice 10-1: Deploying, Loading, and Viewing Data

Deploy Relational Objects

Until now you have designed and configured the logical definitions of your target system. Now you learn how to deploy and create the physical instance of your target.

In this practice, you deploy the relational and dimensional objects. You then deploy the ETL mapping. Finally, you deploy and execute the process flow.

- 1) Launch the Control Center Manager.

Register the Locations

- 2) Register the following locations:

Location Name	Schema User Name	Password
SALES_WH LOCATION	SALES_WH	SALES_WH
XSALES LOCATION	XSALES	XSALES
OWF_MGR_LOCATION	OWF_MGR	OWF_MGR

You are now ready to deploy the objects.

Deploy Sequences

- 3) Deploy the sequences in the SALES_WH target module.

Deploy Tables

- 4) Deploy the Tables in the SALES_WH target module.

You have deployed the Tables objects. You now deploy the dimensional objects. Before you can deploy the cube, you should deploy the related dimensions.

Deploy Dimensional Objects

- 5) Deploy the dimensions in the SALES_WH target module.

Deploy the Cube

- 6) Deploy the cube in the SALES_WH target module.

- 7) Deploy the mappings in the SALES_WH target module.

- 8) Deploy the LOAD_SALES_WH process flow. Expand OWF_MGR_LOCATION to find the LOAD_SALES_WH process flow.

- 9) Execute the LOAD_SALES_WH process flow.

- 10) View the execution details.

- 11) Close the Control Center Manager.

Using the data viewer, you can view the data loaded into the dimensions and the cube from inside the OWB design client environment.

- 12) View the CHANNELS dimension data.

- 13) Close the Dimension Data Viewer.
- 14) View the SALES cube data.
- 15) Close the Cube Data Viewer.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

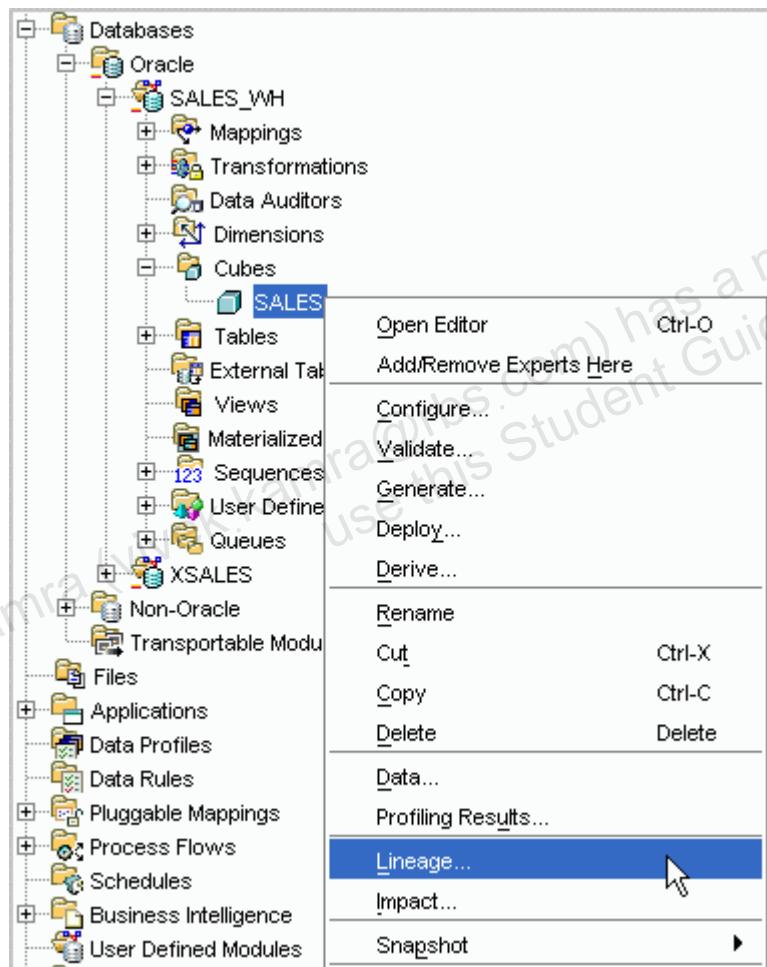
Practice 10-2: Metadata Management

The Warehouse Builder Dependency Manager enables you to determine the impact of the changes made to the object definitions or the metadata in the Warehouse Builder repository. You also learn how to use the Oracle Warehouse Builder Repository Browser, an HTML-based interface to view and generate reports on all repository metadata objects and the relationships between those objects.

The Metadata Dependency Manager generates lineage and impact diagrams for any data object. A lineage diagram traces the process flows for an object back to the data source and displays all objects along that path. An impact diagram identifies all the objects that are derived from selected object.

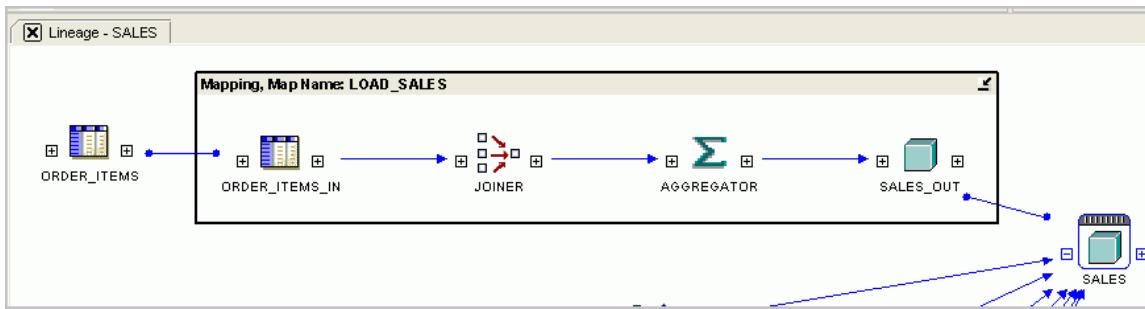
Lineage Analysis for the SALES Cube

- 1) Launch the Metadata Dependency Manager for a lineage analysis of the SALES cube.



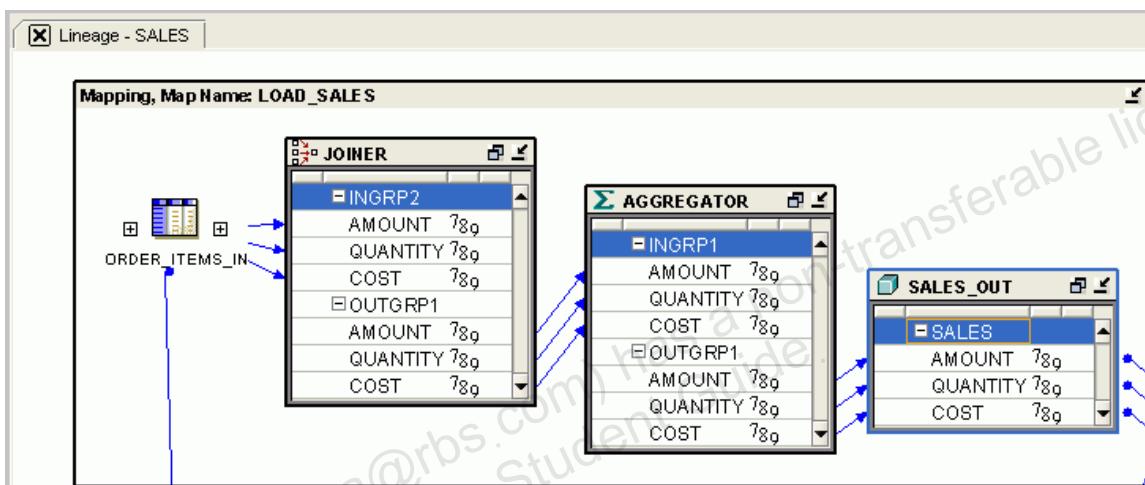
- 2) Find out where all the information in the SALES cube comes from.
- 3) What is the connection between the ORDER_ITEMS table and the SALES cube?

Practice 10-2: Metadata Management (continued)



Drill Down to Attribute Level Impact Analysis

- 4) Find out which attributes are being processed by the aggregator operator.



- 5) Close the Metadata Dependency Manager window.

Impact Analysis for the SALES Cube

- 6) Launch the Metadata Dependency Manager for an impact analysis of the SALES cube.
- 7) Close the Metadata Dependency Manager.

Administering the Warehouse Using Repository Browser

Start the OWB Browser Listener

- 8) Start the OWB Browser Listener. Minimize the Start OWB Browser Listener window after you see that it has successfully started the browser listener.

Note: What you are working on in this classroom setup is a standalone version of the Repository Browser. In your working environment, you need to run one instance of the OWB Browser Listener in your enterprise. Then anyone with the login credentials and a Web browser can access OWB repository reports, without the need to have the OWB Client installed.

Start the Repository Browser

- 9) Launch the Warehouse Builder Repository Browser.

Practice 10-2: Metadata Management (continued)

10) In the Browser window, enter the connection information details as:

Username: rep_owner
Password: rep_owner
Host address: localhost
Host Port Number: 1521
Host Service Name: orcl

Note: The Control Center of the Repository Browser provides the following three types of reports: Deployment Reports, Execution Reports, and Management Reports. Also available is the Design Objects reports.

Design Objects Reports

- 11) List the objects in the SALES_WH module. For security reasons, you will be prompted for the connection details. Provide the same details that you specified at the time of connection in step 10 of this practice.
- 12) Open the Control Center Reports page.

The screenshot shows the Oracle Repository Browser Control Center Reports page. At the top left is a 'Navigator' button. Below it is a 'Search' section with a 'Search By Type' dropdown menu. A red dashed circle highlights this dropdown. To its right is a search input field and a 'Search' button. Below the search section is a 'All' link. Underneath is a 'Refresh' button. The main area contains a table with the following data:

Focus	Name	Type	Reports	Lineage	Impact
▼	Top			60	60
⊕	rep_owner	Repository		60	60

Below the table is another 'Refresh' button. At the bottom left is a 'Related Links' section with a link to 'Control Center : Reports'. A cursor icon is hovering over this link. At the bottom right are 'Logout' and 'Help' links. The footer of the page states 'Copyright © 2000, 2006, Oracle. All rights reserved.'

Object Summary Report

Practice 10-2: Metadata Management (continued)

13) Examine the object summary report of all the objects.

Reports

Deployment

- [Deployment Schedule Report](#) - to display details of all deployments in time order
- [Object Summary Report](#) - to list all objects with details of their latest deployment
- [Locations Report](#) - to show all locations into which objects have been deployed

Execution

- [Execution Schedule Report](#) - to display details of all execution jobs in time order
- [Execution Summary Report](#) - to list all executable objects with details of their latest execution job

Management

- [Service Node Report](#) - to display and manage service node information for the RAC system

Related Links

 [Design Repository : Navigator](#)

Reports : rep_owner >

Object Summary Report

Repository	Type	Installed Version
rep_owner	RUNTIME REPOSITORY	10.2.0.1.31
Description		

Available Reports

- Deployment Schedule
- Object Summary
- Locations
- Execution Schedule
- Execution Summary

Deployments

Filter on Object Type

Filter on Object Status

Name	Type	Location	Latest Deployment	Obj Status
SALES	Cube	SALES_WH_LOCATION	2006-05-18 16:02:00	Valid
CHANNELS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
CUSTOMERS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
PRODUCTS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
PROMOTIONS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
TIMES	Dimension	SALES_WH_LOCATION	2006-05-18 16:01:40	Valid
LOAD_CHANNELS	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 15:39:52	Valid
LOAD_CUSTOMERS	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 16:03:20	Valid

14) Examine the execution status of the objects.

Execution Jobs

Filter on Type

Filter on Execution Status

Name	Type	Location	Latest Execution	Execution Status	Related Information
LOAD_CHANNELS	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 15:40:05	Complete : OK	 Execution Job  Report
TIMES_MAP	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 15:40:50	Complete : OK	 Execution Job  Report

15) Examine the complete details about the execution of the LOAD_CHANNELS mapping.

16) Log out from the repository browser.

17) Stop the OWB Browser Listener.

Practice 10-2: Metadata Management (continued)

- 18) Click Start > Programs > Oracle–OwbHome10gR2 > Warehouse Builder > Administration > Stop OWB Browser Listener.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice for Lesson 11

Your CLASS_PROJECT project contains a target module SALES_WH that implements the relational data warehouse. The first set of exercises applies to business definitions, enabling the seamless integration with Oracle Discoverer. The second part enables you to perform ad hoc queries on top of relational structures.

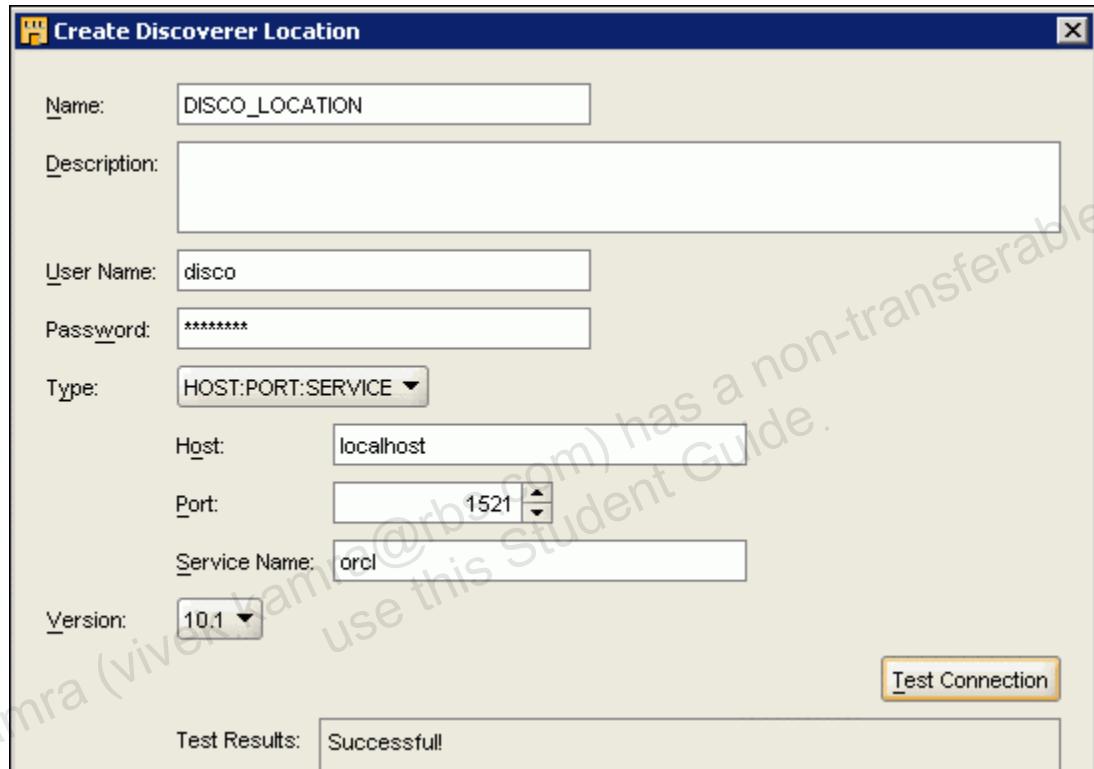
vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice 11-1: Defining Business Intelligence Objects Based on Data Objects

Define a Discoverer Location

A Discoverer location provides details about the system to which the business definitions you create are deployed. This system should have an Oracle Discoverer EUL version 10.1.2 or later installed.

- 1) Define a Discoverer location with the following details:



Define a Business Definition Module

In the previous exercises, you created different kinds of modules—for example, Oracle modules, files module, process flow modules. Modules are storage objects that help you organize objects within a module. Now you create a module to store the business definitions.

- 2) Create a business definition module named DISCOVERER_OBJECTS.
- 3) Locate and examine DISCOVERER_OBJECTS, the Business Definition module that you just created in the Design Center.

Define a Business Area

Business areas contain references to Item Folders stored in your Business Definition module. Warehouse Builder enables you to create a business area to deploy to a Discoverer EUL.

- 4) Create a business area named SALES_DATA.

Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Create a Simple Item Folder

First, you create a simple item folder. Later, you create a complex item folder.

- 5) Create a Simple Item Folder named SALES_PROMOTIONS using the editor. The SALES_PROMOTION item folder is based on the **SALES_WH.PROMOTIONS** table. Connect the columns from the PROMOTIONS table to the item folder.

Note: There are two nodes available for defining item folders. One is at the lower Business Areas level (SALES_DATA). The other is at the higher Business Definition Module level (DISCOVERER_OBJECTS). For this example, you create an item folder at the higher module level.

- 6) Save your work, and close the editor.

Create a Drill Path

A drill path primarily defines a hierarchy relationship in a dimension. Warehouse Builder enables you to create a drill path to deploy to a Discoverer EUL.

Besides creating drill paths manually, you can use a wizard to create Drills to Detail. Alternately, if you derive a dimension, you get the drill path automatically.

- 7) Create a new drill path, PROMOTIONS_DRILL, under the DISCOVERER_OBJECTS business definition module. Create a drill path, with drill levels in the following order:

Drill Level	Item Folder	Drill Level Item
TOTAL	SALES_PROMOTIONS	TOTAL_NAME
SUB_CATEGORY	SALES_PROMOTIONS	SUBCATEGORY_NAME
CATEGORY	SALES_PROMOTIONS	CATEGORY_NAME

- 8) Save your work.

Derive Individual Objects

You have now created a number of structures for Discoverer manually. In previous lessons, you defined a large amount of metadata as data object definitions. Rather than manually adding more item folders and drill paths, you can reuse your metadata definitions and propagate them into business definitions.

With Oracle Warehouse Builder 10g Release 2, you can derive sets of individual data objects or a grouping of data objects in the form of a collection. The following steps guide you through both scenarios.

- 9) Derive the dimensions and the cube from the SALES_WH target module using the Perform Derivation Wizard.
- 10) Open the SALES_DATA business area. Verify that the tree shows the definitions that were derived.

Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Warehouse Builder creates hidden Simple Item Folders on top of the relational tables that implement the dimensions and the cube, and it creates Complex Item Folders on top of the Simple Item Folders, which are visible to users. The derivation process also creates the drill paths that implement the respective hierarchies on the dimensions.

Finally, because you selected a business area as a target, all item folders are referenced by that business area.

Derive Objects from a Collection

You just derived business objects from five dimensions and a cube. In addition to deriving from individual object definitions, you can derive from OWB “collections.” You create a collection, and then derive business objects from that collection.

Create a New Business Definitions Module

- 11) Create a new Business Definitions module named MART_DEFINITIONS. On the wizards Connection Information page, use DISCO_LOCATION.



Create a Collection

- 12) Create a collection named SALES_MART. The collection refers to two dimensions and one cube, namely, the CHANNELS and PRODUCTS dimensions and the SALES cube.

Derive Business Definitions from the SALES_MART Collection

- 13) Derive business definitions from the SALES_MART collection. Select the MART_DEFINITIONS business definition module to contain the derived objects.
- 14) Examine the objects created in SALES_MART.
- 15) Create a complex folder named PRODUCT_SALES. Select PRODUCTS and SALES as the source items
- 16) Open the PRODUCT_SALES item folder in the Data Object Editor.
- 17) For the AMOUNT item, choose AVG as the default aggregation.
- 18) Add an item named PROFIT, with the “Visible to user” option selected.
- 19) Create a formula for this newly created PROFIT item. The formula is:
PRODUCT_SALES.AMOUNT – PRODUCT_SALES.COST

Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

- 20) Save your work.
- 21) Close the Data Object Editor.

Create a List of Values

Your manager has an additional requirement: Show a list of values for all products, enabling the report to be restricted based on one or more items selected from the list of products. To implement this requirement, create a list of values.

- 22) Create a list of values named PRODUCTS_LOV. Select PRODUCTS.PRODUCT_NAME as the item that defines the list of values. Select PRODUCTS_SALES.PRODUCT_NAME as the item that references the list of values.

Create “Drill to Detail”

Now your manager wants to drill down to detail on the specifics of the product. To enable this functionality for an item, you can create Drill to Detail and relate the item to that Drill to Detail.

Currently, each of the following constructs materializes as an item class in Discoverer:

- List of Values
- Drill to Detail
- Alternative Sort

Discoverer allows only a maximum of one item class per item. Therefore, to enable Drill to Detail on the item class that implements your PRODUCTS_LOV list of values, you need to edit the PRODUCTS_LOV list of values.

- 23) Edit PRODUCTS_LOV to select the Set As Drill to Detail option.

- 24) Save your work.

Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer

In the previous practice, you defined business intelligence objects for Discoverer. In this practice, you use Discoverer Administrator to create an End User Layer (EUL), deploy the OWB business intelligence objects to the End User Layer schema, and then use Discoverer Plus Relational to analyze the data.

In Discoverer Administrator, Create an EUL in the DISCO Database User

- 1) Using Oracle Discoverer Administrator, create an EUL named DISCO.

Deploy BI Objects to the End User Layer

You deploy the BI objects to the End User Layer.

- 2) Register DISCO_LOCATION.
- 3) Deploy DISCOVERER_OBJECTS to the EUL.
- 4) Grant select privileges to DISCO on tables in SALES_WH.

Log In to Discoverer Plus Relational as EUL_FROM_OWB and Create a Worksheet

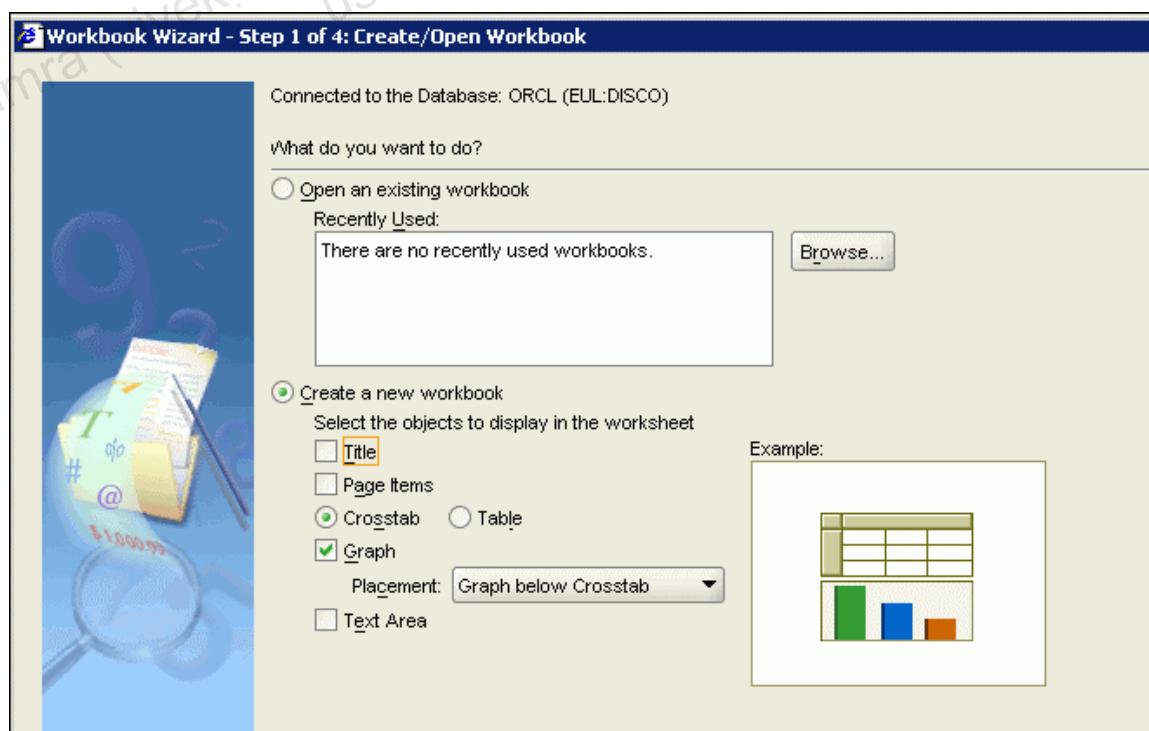
- 5) Open OracleBI Discoverer to create a worksheet from the populated EUL.

Log In to Discoverer Plus Relational as DISCO and Create a Worksheet

If you receive a JInitiator Security Warning pop-up message, click “Grant this session.”

The Workbook Wizard is launched.

- 6) On the Workbook Wizard, select Crosstab and Graph and deselect all other check boxes.



Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)

- 7) On the Select Items page, select the following: PRODUCTS.Product_Name, CHANNELS.Channel_Name, and SALES CUBE.Sales Amount.

- 8) On the Crosstab Layout page, arrange the items as shown in the screenshot.

Make sure that you have at least one side item to run down the left. Click Finish. Now you can analyze the data.

Product_Name	Channel_Name	Sales Amount SUM		
		► Direct Sales	► Internet	► Partners
► 1.44MB External 3.5" Diskette		138170.41	22255.76	60503.14
► 128MB Memory Card		315483.78	89251.24	169500.41
► 17" LCD w/built-in HDTV Tuner		4457699.64	1058952.3	1692366.8
► 18" Flat Panel Graphics Monitor		3038192.49	1092383.31	1127568.55
► 256MB Memory Card		394743.84	71203.21	127465.48
► 3 1/2" Bulk diskettes, Box of 100		251148.22	31545.15	89103.81
► 3 1/2" Bulk diskettes, Box of 50		158808.74	26181.25	70499.44
► 5MP Telephoto Digital Camera		3400862.36	1381685.55	1561476.75
► 64MB Memory Card		13839.95	4767.1	9489.93
► 8.3 Minitower Speaker		1736923.68	1007545.07	1107740.14
► Adventures with Numbers		114737.73	18077.88	43456.27
► Bounce		125650.53	38860.01	81562.42
► CD-R Mini Discs		259168.4	37037.77	89793.5
► CD-R with Jewel Cases, pACK OF 12		99958.63	20436.87	55277.7
► CD-R, Professional Grade, Pack of 10		108998.31	17901.45	44458.22
► CD-RW, High Speed Pack of 5		144882.36	23670	59975.9
► CD-RW, High Speed, Pack of 10		69847.16	12010.24	25233.48
► Comic Book Heroes		76484.91	8671.24	16494.27
► DVD-R Disc with Jewel Case, 4.7 GB		311400.09	131909.96	136008.46
► DVD-R Discs, 4.7GB, Pack of 5		580120.97	99509.09	229076.03

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Appendix B

Practice Solutions

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Table of Contents

Solutions for Practice 1-1: Create a Repository Owner and User	4
Solutions for Practice 1-2: Invoking OWB and Opening a Project.....	9
Solutions for Practice 2-1: Importing Metadata.....	16
Solutions for Practice 3-1: Create a Flat File Source Module	21
Solutions for Practice 3-2: Create a Relational Database Source Module.....	28
Solutions for Practice 4-1: Profile the Data	36
Solutions for Practice 4-2: View and Analyze the Profile Results	43
Solutions for Practice 5-1: Create an External Table.....	56
Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target	65
Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings..	83
Solutions for Practice 7-1: Create a Dimension Using the Wizard	97
Solutions for Practice 7-2: Create a Dimension Using the Editor	103
Solutions for Practice 7-3: Create a TIMES Dimension Using the Time Wizard	112
Solutions for Practice 7-4: Create a Cube Using the Editor	117
Solutions for Practice 7-5: Design ETL Mappings.....	123
Solutions for Practice 8-1: Handling Slowly Changing Dimensions.....	143
Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow.....	151
Solutions for Practice 9-2: Create a Schedule.....	164
Solutions for Practice 10-1: Deploying, Loading, and Viewing Data	177
Solutions for Practice 10-2: Metadata Management.....	187
Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects	200
Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer.....	230

Practice Solutions for Lesson 1

This practice is to enable you to invoke Oracle BI Warehouse Builder Design Center and create a repository owner and repository user by using the Logon window.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 1-1: Create a Repository Owner and User

Before logging in to the Design Center, you must create a user with which to log in. You can create users with the Get Started button of the Design Center Logon dialog box. You create a repository user and a highly privileged repository owner. Then you install the OWB repository.

Using the OWB Design Center Get Started Button to Create a Repository User and Owner

- 1) Invoke the Oracle Warehouse Builder Design Center by double-clicking the OWB10gR2 Design Center icon on the desktop.

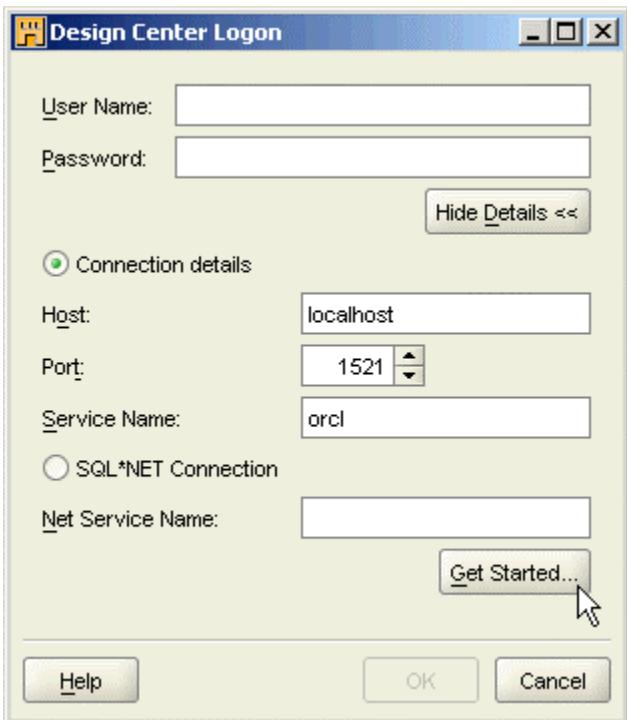
Answer:

From the Windows Start menu, select Programs > Oracle – OwbHome10gR2 > Warehouse Builder > Design Center. The Design Center Logon window appears. Or, double-click the OWB10gR2 Design Center icon on the Desktop.

- 2) In the Design Center Logon window, click the Get Started button. Click the Show Details button if you are not able to see the Get Started button.



Solutions for Practice 1-1: Create a Repository Owner and User (continued)



Answer:

1. If the Design Center Logon window contains a **Show Details** button, click it to expand the window.
2. Click the **Get Started** button.

The Repository Assistant Welcome page appears.

- 3) Click Next on the Repository Assistant Welcome page to launch the Install Type page.

Answer:

On the Repository Assistant Welcome page, click **Next**. The Install Type page appears.

- 4) Select Basic Install on the Install Type page, and click Next to launch the Repository User and Connection Information page.

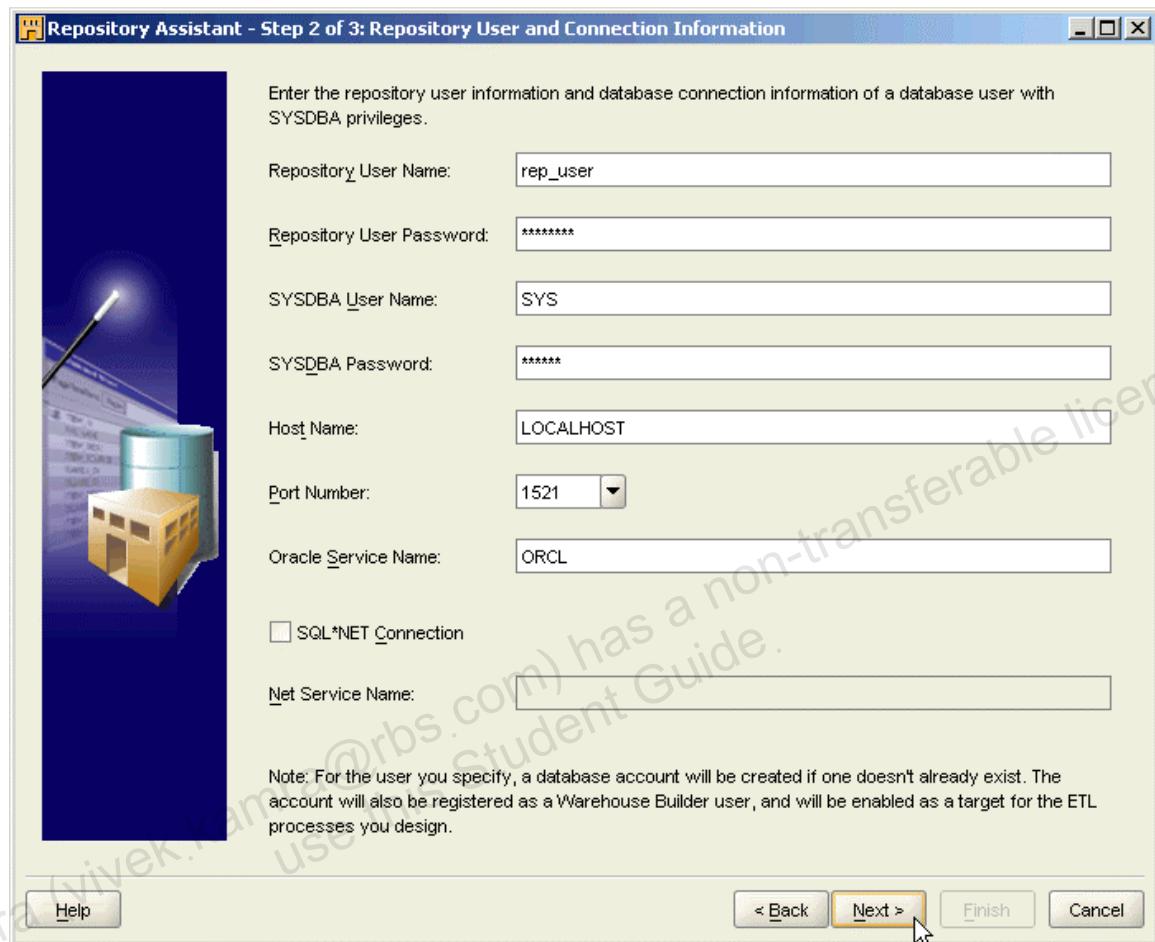
Answer:

On the Repository Assistant Basic Install page, select **Basic Install**, and click Next. The Repository User and Connection Information page appears.

- 5) On the Repository User and Connection Information page, create a new user **rep_user** with the following details:
 - a) Repository User Name: **rep_user**
 - b) Repository User Password: **rep_user**
 - c) SYSDBA User Name: **SYS**
 - d) SYSDBA Password: **oracle**

Solutions for Practice 1-1: Create a Repository Owner and User (continued)

- e) Host Name: **LOCALHOST**
- f) Port Number: **1521**
- g) Oracle Service Name: **ORCL**



Answer:

1. On the Repository User and Connection Information page, enter the following information to create the new user named **rep_user**:
 - a. Repository User Name: **rep_user**
 - b. Repository User Password: **rep_user**
 - c. SYSDBA User Name: **SYS**
 - d. SYSDBA Password: **oracle**
 - e. Host Name: **LOCALHOST**
 - f. Port Number: **1521**
 - g. Oracle Service Name: **ORCL**
2. Click **Next**. The Password Confirmation dialog box pops up.
- 6) In the Password Confirmation dialog box, confirm the password, and click **OK**.

Solutions for Practice 1-1: Create a Repository Owner and User (continued)

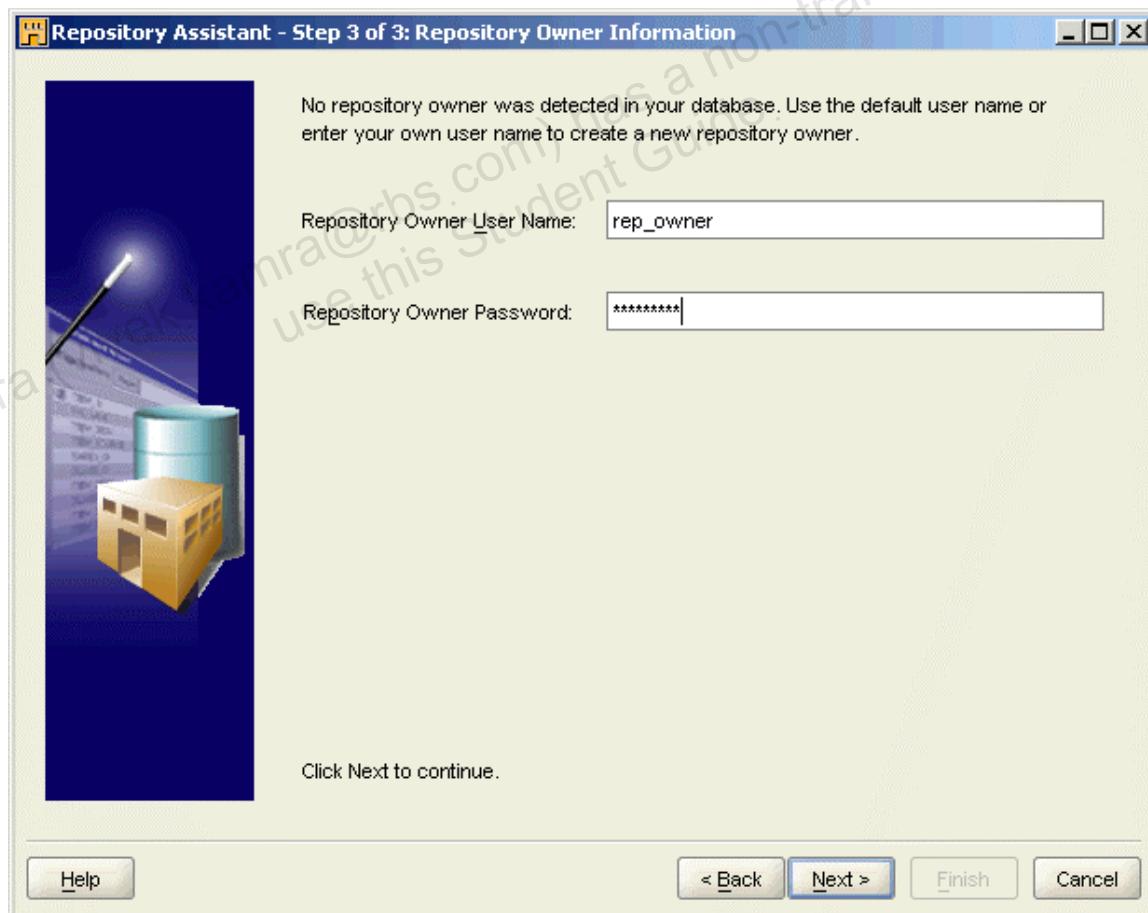
- a) Re-enter Password: **rep_user**

Answer:

1. In the Password Confirmation dialog box, enter the following information:
Re-enter Password: **rep_user**
 2. In the Password Confirmation window, click OK.
- 7) On the Repository Owner Information page, create the owner named **rep_owner**. In the Password Confirmation dialog box, reenter the password.

On the Repository Owner Information page, if no owner has yet been defined, Warehouse Builder presents a default owner name (OWB_10_2_0_1_5) based on the version of Warehouse Builder installed. You instead provide a different owner name. The repository owner is a highly privileged database user with access to additional security features.

- a) Repository Owner User Name: **rep_owner** (type over the default)
- b) Repository Owner Password: **rep_owner**



Note: If someone has previously created another repository owner on this computer, you will be unable to type over the existing name in the Repository Owner User Name field. In that case, click Cancel on the Repository Owner Information page, and

Solutions for Practice 1-1: Create a Repository Owner and User (continued)

create rep_owner by invoking the Repository Assistant as follows: Select Start > Programs > Oracle – OwbHome10gR2 > Warehouse Builder > Administration > Repository Assistant. Choose Advanced Setup and follow the prompts to create a new repository owner, naming it rep_owner. Then rerun steps 1 through 5. In step 6, choose rep_owner.

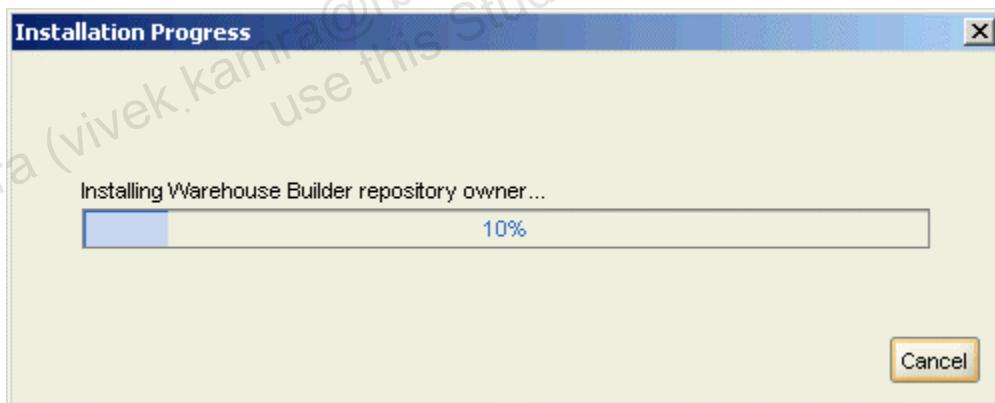
Answer:

1. On the Repository Owner Information page, enter the following information to create the owner named rep_owner:
 - a. Repository Owner User Name: rep_owner (type over the default)
 - b. Repository Owner Password: rep_owner
2. In the Password Confirmation dialog box:
 - a. Reenter **rep_owner** in the Re-enter Password field.
 - b. Click OK.
- 8) On the Summary page, examine the information, and click Finish.

Answer:

On the Summary page, the information displays the values selected on the previous wizard pages. Review the information, and click Finish.

- 9) An Installation Progress window appears. Installation of the repository owner will take a number of minutes. Your instructor may resume lecturing during this time.



- 10) An Installation Successful window appears eventually. Click OK. The Repository Assistant closes. If the Design Center Logon window remains open, click Cancel to close it. In the Warehouse Builder Warning window, click Yes to quit Warehouse Builder. You will log in later in this lesson.

Solutions for Practice 1-2: Invoking OWB and Opening a Project

In this practice, you invoke the Warehouse Builder Design Center, log in as the user, open the existing project named “My Project,” create a new project, and examine the project’s properties.

- 1) Start the OWB Design Center by using the Desktop icon or the Windows Start button.

Answer:

Double-click the OWB desktop icon; or, from the Windows Start menu, select Programs > Oracle – OwbHome10gR2 > Warehouse Builder > Design Center to invoke Warehouse Builder. The Design Center Logon window appears.

- 2) Click **Show Details** to expand the Design Center Logon window. Enter the connection details as:

- a) Host: **localhost**
- b) Port: **1521**
- c) Service Name: **orcl**

In the top portion of the Design Center Logon window, enter the username and password that you created in the previous practice:

- a) User Name: **rep_user**
- b) Password: **rep_user**



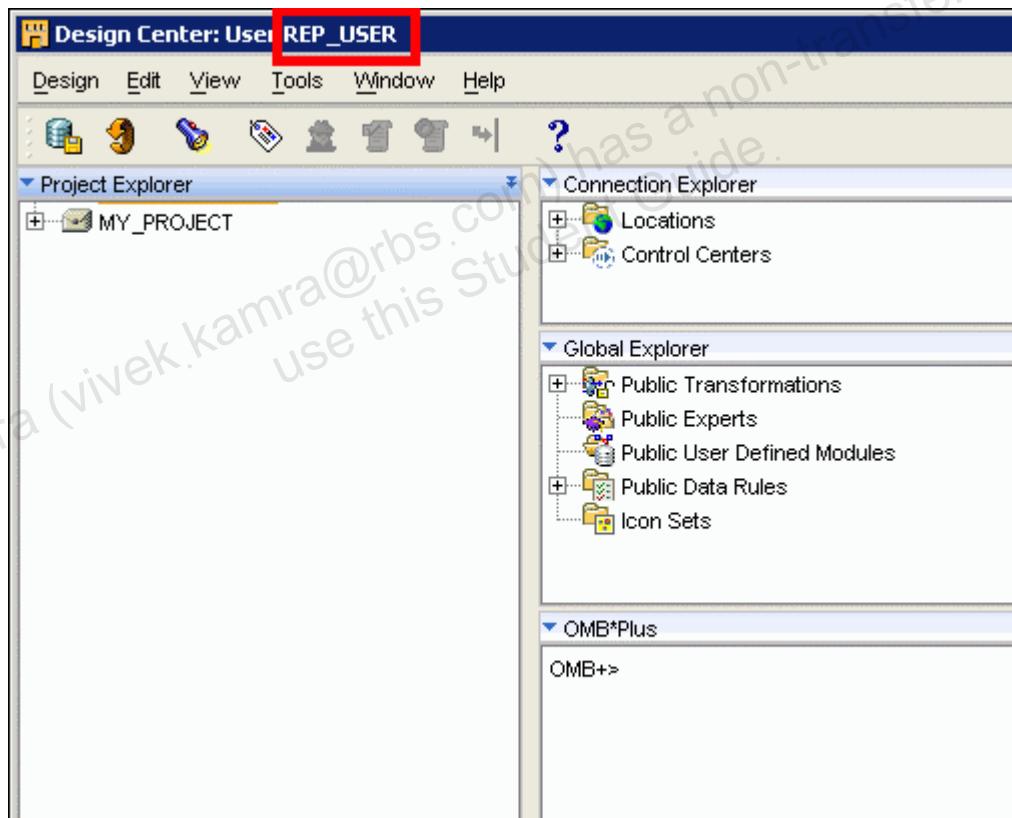
Answer:

1. Before logging in with the username and password, you must provide connection information. Click Show Details.

Solutions for Practice 1-2: Invoking OWB and Opening a Project (continued)

2. The Design Center Logon window expands. Provide the host name, port number, and service name that were defined during OWB installation on your classroom computer.
 - a. Host: localhost
 - b. Port: 1521
 - c. Service Name: orcl
3. In the top portion of the Design Center Logon window, enter the username and password that you created in the previous practice.
 - d. User Name: rep_user
 - e. Password: rep_user

After you have entered all the connection details in the Design Center window, click OK. The Design Center window appears.



Creating a New Project with the Create Project Wizard

The Design Center is divided into three panels: Project Explorer, Connection Explorer, and Global Explorer. In the Project Explorer, you see MY_PROJECT, an empty default project created when Warehouse Builder is installed.

- 3) In the Design Center, select New from the Design menu. The Create Project dialog box appears. Enter the following information:

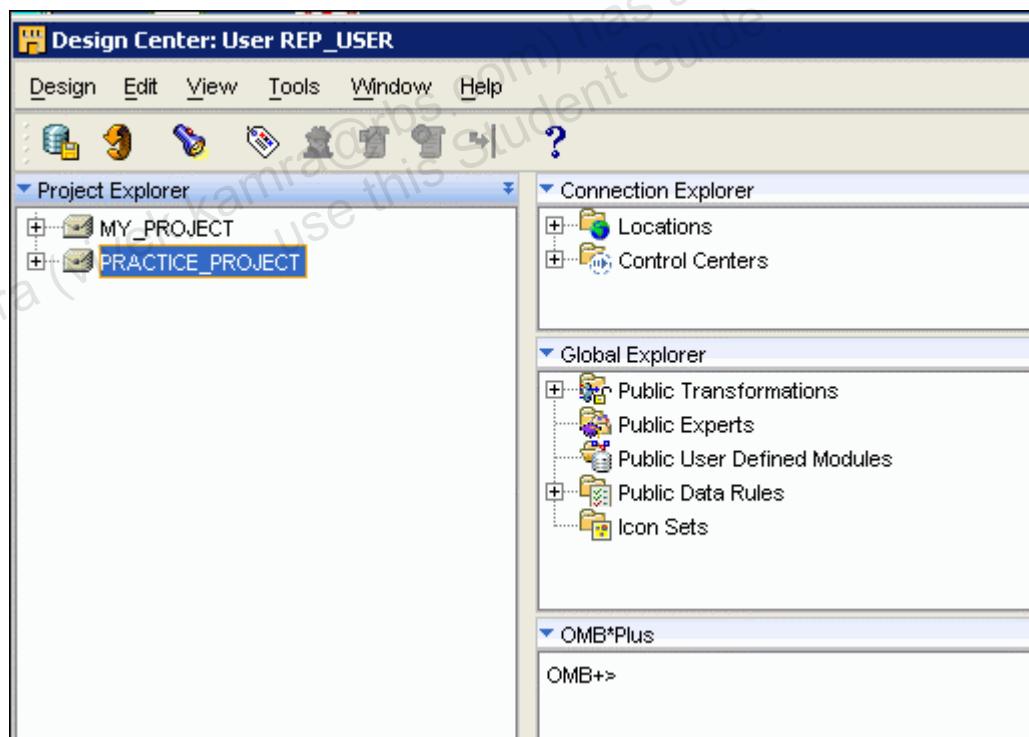
Solutions for Practice 1-2: Invoking OWB and Opening a Project (continued)

- a) Name: Practice_Project
- b) Description: Project for exploring the OWB client user interface

Notice that your new project is listed in the Project Explorer.

Answer:

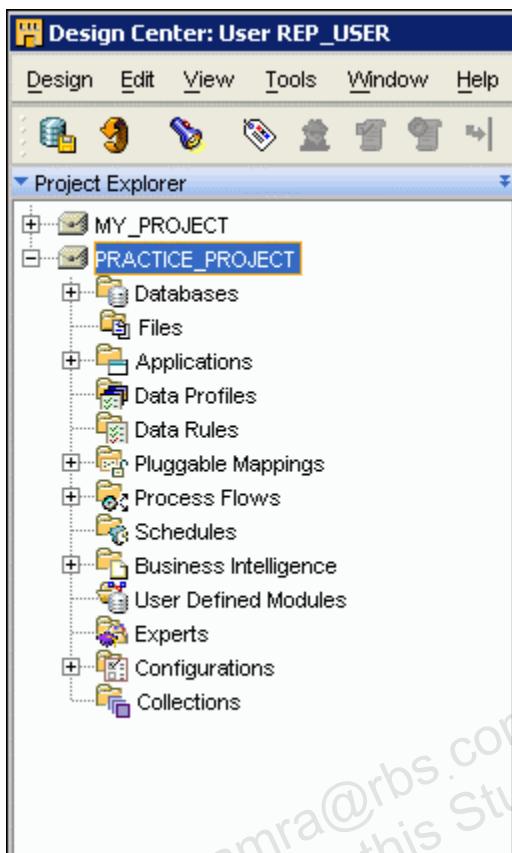
1. In the Design Center, select New from the Design menu. The Create Project dialog box appears.
2. In the Create Project dialog box, enter the following information:
 - a. Name: Practice_Project
 - b. Description: Project for exploring the OWB client user interface
3. In the Create Project dialog box, click OK. You notice that the project (PRACTICE_PROJECT) that you just created is listed in the Project Explorer.
- 4) Examine all the nodes in the PRACTICE_PROJECT project tree.



Answer:

1. In the Project Explorer, click PRACTICE_PROJECT to select it (if it is not already selected).
2. In the Design Center, from the View menu, select Expand All. Examine all the nodes in the tree.

Solutions for Practice 1-2: Invoking OWB and Opening a Project (continued)



- 5) After examining the entire node, collapse the tree view by selecting Collapse All from the View menu.

Answer:

After examining all the nodes, in the Design Center, select Collapse All from the View menu.

Viewing Project Properties

- 6) Check the session properties by using the Session Properties option from the Help menu.

Answer:

In the Design Center, select Session Properties from the Help menu. This is one way to check the username and other connection settings for your current OWB session. Click Close.

Viewing Project Preferences

- 7) Examine the default User Preference by selecting the Preferences option from the Tools menu. In the Preferences window, click each node on the left and examine the value on the right. **Note:** For now, do not change any settings. Later in the course, you learn more about changing these settings as and when required.

Solutions for Practice 1-2: Invoking OWB and Opening a Project (continued)

Answer:

1. In the Design Center, select Preferences from the Tools menu.
2. Examine the settings by clicking each node on the left. Do not change any settings. **Note:** For now, do not change any settings. Later in the course, you will learn more about changing these settings as and when required.
3. Click OK.

Deleting the Practice Project

- 8) Delete the project (PRACTICE_PROJECT) you just created.

Answer:

In the Design Center, right-click **PRACTICE_PROJECT**, and select **Delete**. Click **OK** in the Warehouse Builder Warning dialog box. Do not exit from OWB.

Note: This warning dialog box pops up every time you delete a project. Click OK and carry on with your work.

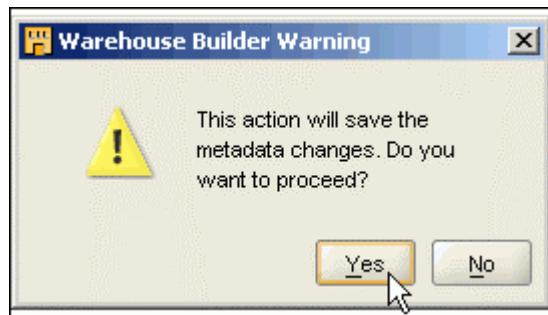
Saving Your Work

- 9) Save your work using the Save All icon on the toolbar, or the Save All option from the Design menu.



Answer:

1. In the Design Center, click the **Save All** icon on the toolbar, or select the **Save All** option from the **Design** menu. Click **Yes** in the Warehouse Builder Warning dialog box. **Note:** The warning dialog box pops up every time you click Save All. Click OK and continue with your work.



- 10) After saving your work, exit OWB.

Answer:

**Solutions for Practice 1-2: Invoking OWB and Opening a Project
(continued)**

After having saved your work, select **Exit** from the **Design** menu. Click **Yes** in the Warehouse Builder Warning dialog box. Every time you exit OWB, this warning dialog box is displayed.

Practice Solutions for Lesson 2

In this practice, you log in to the Design Center as rep_owner and use the Metadata Import dialog box to import Warehouse Builder metadata contained in the lab2.mdl export file. The partially predefined CLASS_PROJECT is created and used in this course's remaining lessons.

Solutions for Practice 2-1: Importing Metadata

You create the partially predefined project CLASS_PROJECT by using MDL Import.

Log in to the Warehouse Builder Design Center.

- 1) Log in to the Warehouse Builder Design Center with the following details:

User Name: **rep_owner**

Password: **rep_owner**

Answer:

1. Double-click the OWB10g R2 Design Center Desktop icon; or, from the Windows Start menu, select Programs > Oracle – OwbHome10gR2 > Warehouse Builder > Design Center to invoke the Warehouse Builder Design Center. The Design Center Logon window appears.
2. Enter the username and password that you created in Practice 1:

User Name: rep_owner

Password: rep_owner

3. Click OK.

Security Settings

For the purpose of this course, you change the value for “Persist location password in metadata” security parameter. This eliminates the need for you to repeatedly provide passwords.

- 2) In the OMB*Plus panel, enter the following command:

```
OMBALTER USER 'REP_OWNER' SET PROPERTIES \
(DESIGNREPOS_PWD_PERSIST) VALUES('true')
```

Answer:

4. At the OMB Plus prompt, in the OMB Plus panel, enter the following:

```
OMB+> OMBALTER USER 'REP_OWNER' SET PROPERTIES \
(DESIGNREPOS_PWD_PERSIST) VALUES('true')
```

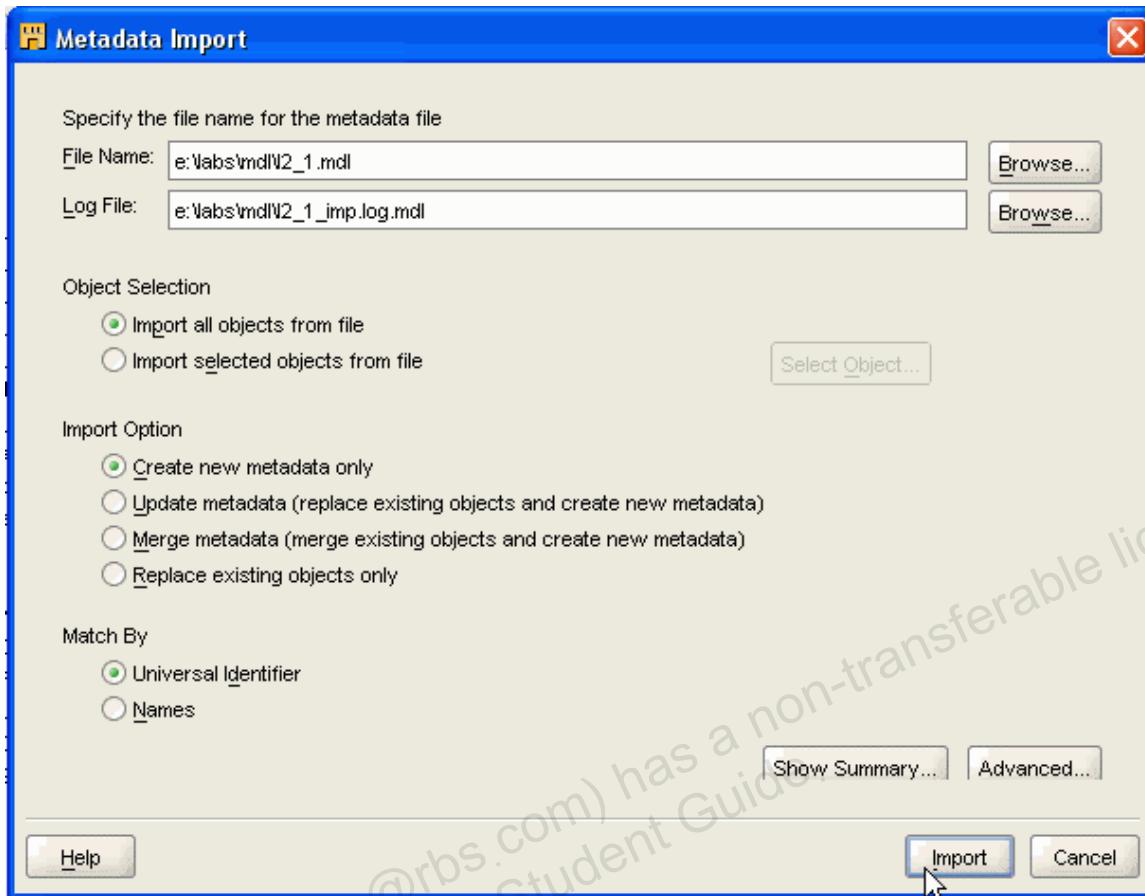
```
OMB+> OMBCOMMIT
```

You now import the metadata that will create the partially predefined project.

Create the Metadata Using MDL Import

- 3) Launch the Metadata Import Utility, and import the **e:\labs\mdl\L2_1.mdl** file.

Solutions for Practice 2-1: Importing Metadata (continued)



Answer:

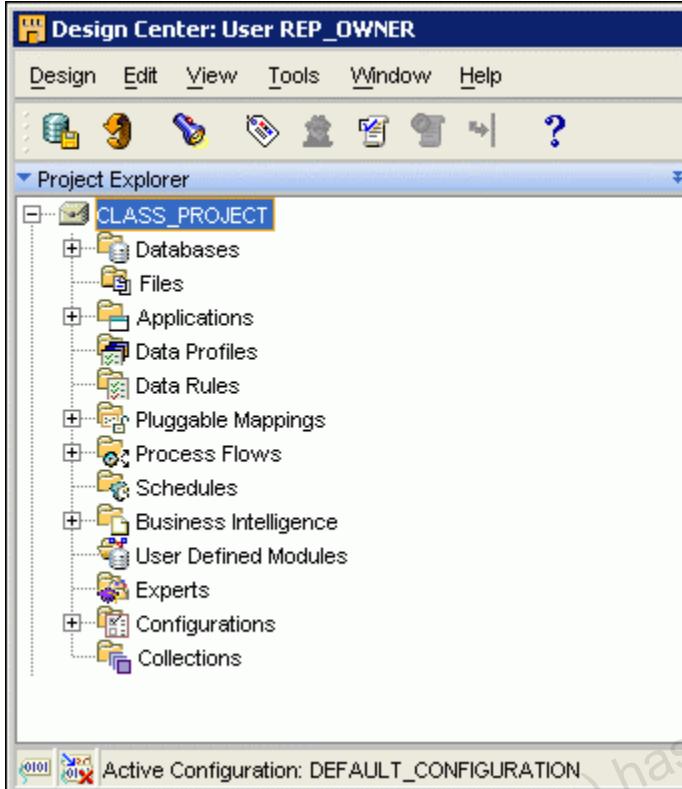
1. In the Design Center, select Import > Warehouse Builder Metadata from the Design menu. The Metadata Import dialog box appears.
2. In the Metadata Import dialog box, enter the following:
File Name: **e:\labs\mdl\l2_1.mdl**
Accept the default option, and click **Import**.
3. The Metadata Import Progress dialog box displays the progress of the import.
When the import completes successfully, click Close.

Examine the CLASS_PROJECT Metadata

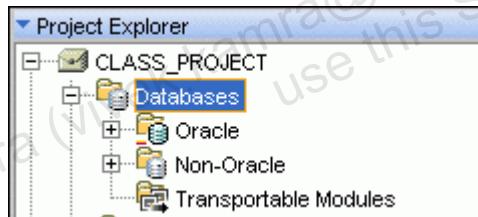
A project is a container to manage your design work. Warehouse Builder contains wizards, object editors, property sheets, and object-finding tools that assist you in designing your business intelligence system.

- 4) In the Design Center, in the Project Explorer, expand CLASS_PROJECT. Various object types appear in the tree: Databases, Files, Applications, Data Profiles, and so on.

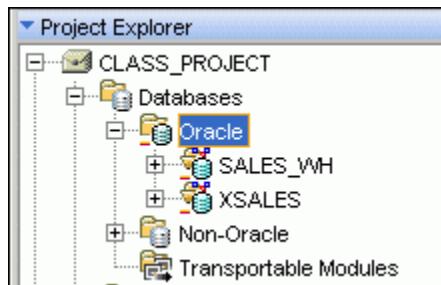
Solutions for Practice 2-1: Importing Metadata (continued)



- 5) Expand the Databases node, and you see various object types that can participate in your design: Oracle, Non-Oracle, and Transportable Modules.

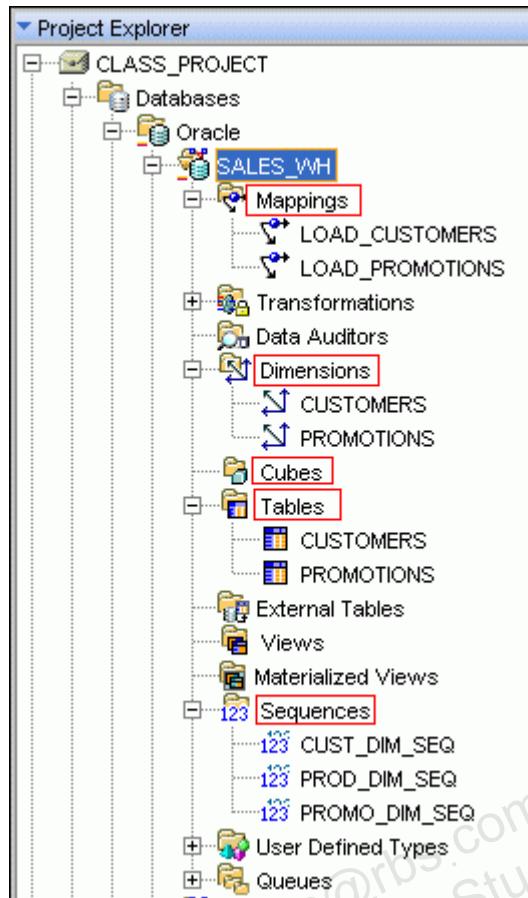


- 6) Expand the Oracle node, and you see two modules predefined by the setup scripts: SALES_WH and XSALES. Modules are logical groupings of source or target definitions.



- 7) Expand the SALES_WH module. Various object types appear in the tree: Mappings, Transformations, Dimensions, Cubes, Tables, and so on. Expand some of these nodes to see the objects that have been predefined by the setup scripts.

Solutions for Practice 2-1: Importing Metadata (continued)



Practice Solutions for Lesson 3

This lesson has two parts. In part 1, you create a flat file source module. In part 2, you create an Oracle database module.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 3-1: Create a Flat File Source Module

In this practice, you create a flat file source module. The flat file module stores the definitions of the flat files. After you create the flat file module, you import metadata from a flat file by using the Import Metadata Wizard.

Create a Source Module

- 1) Create a flat file source module by using the Create Module Wizard, with the following details:

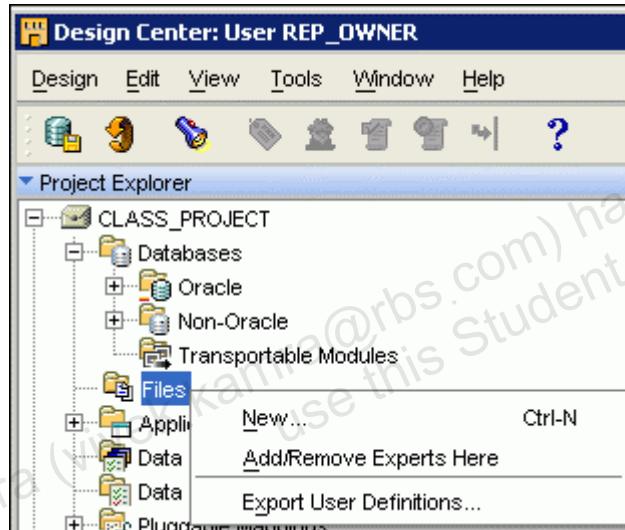
Module name: **FILE_SOURCE**

Description: **This is a Flat file source module for the Geography.txt file**

Change the location name to **FILE_SOURCE_LOCATION**.

File system location: **e:\labs\sourcefiles\GEOGRAPHY.txt**

On the Connection Information page, ensure that “Import after finish” is selected.



Answer:

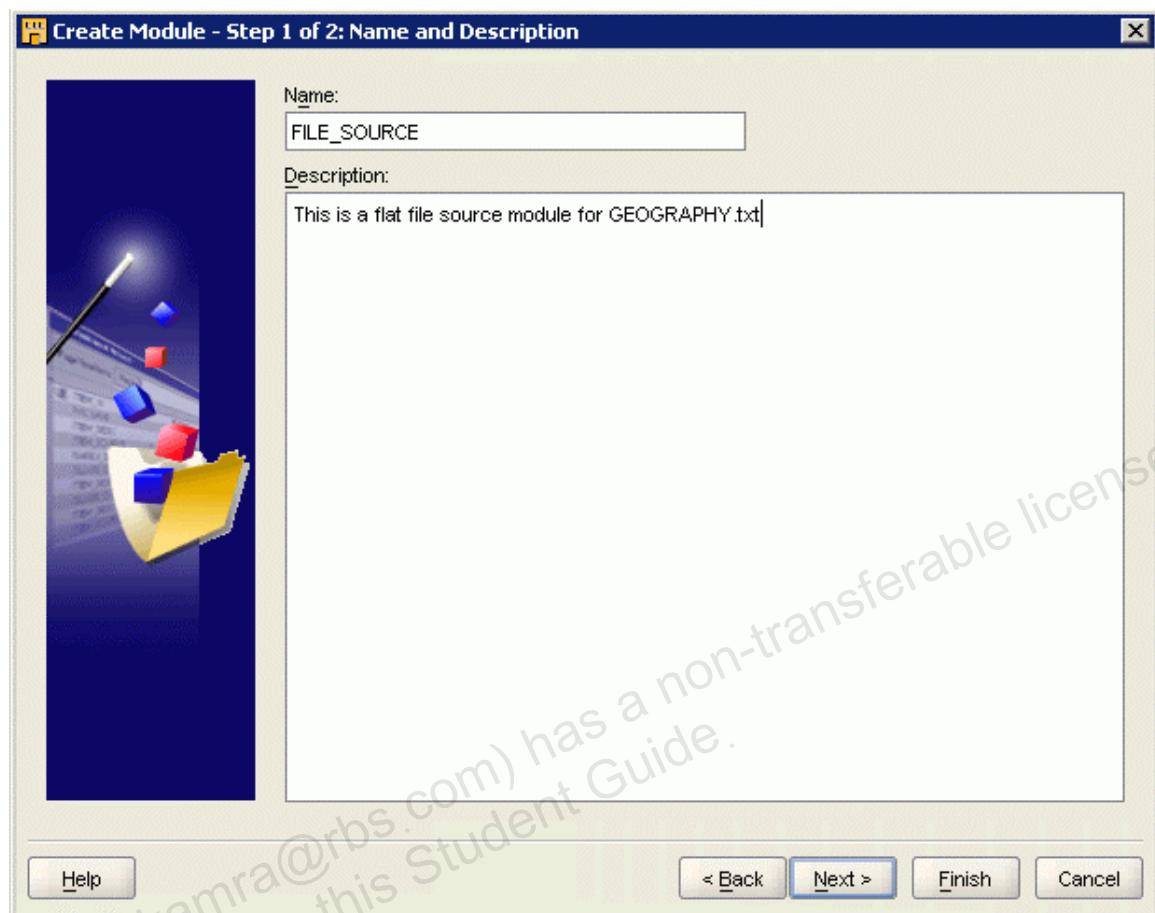
1. In the Project Explorer, expand CLASS_PROJECT. Right-click the **Files** node. Select **New** to open the Create Module Wizard. Click **Next** on the Welcome page.
2. On the Name and Description page, enter the following information:

Name: **FILE_SOURCE**

Description: This is a Flat file source module for the Geography.txt file

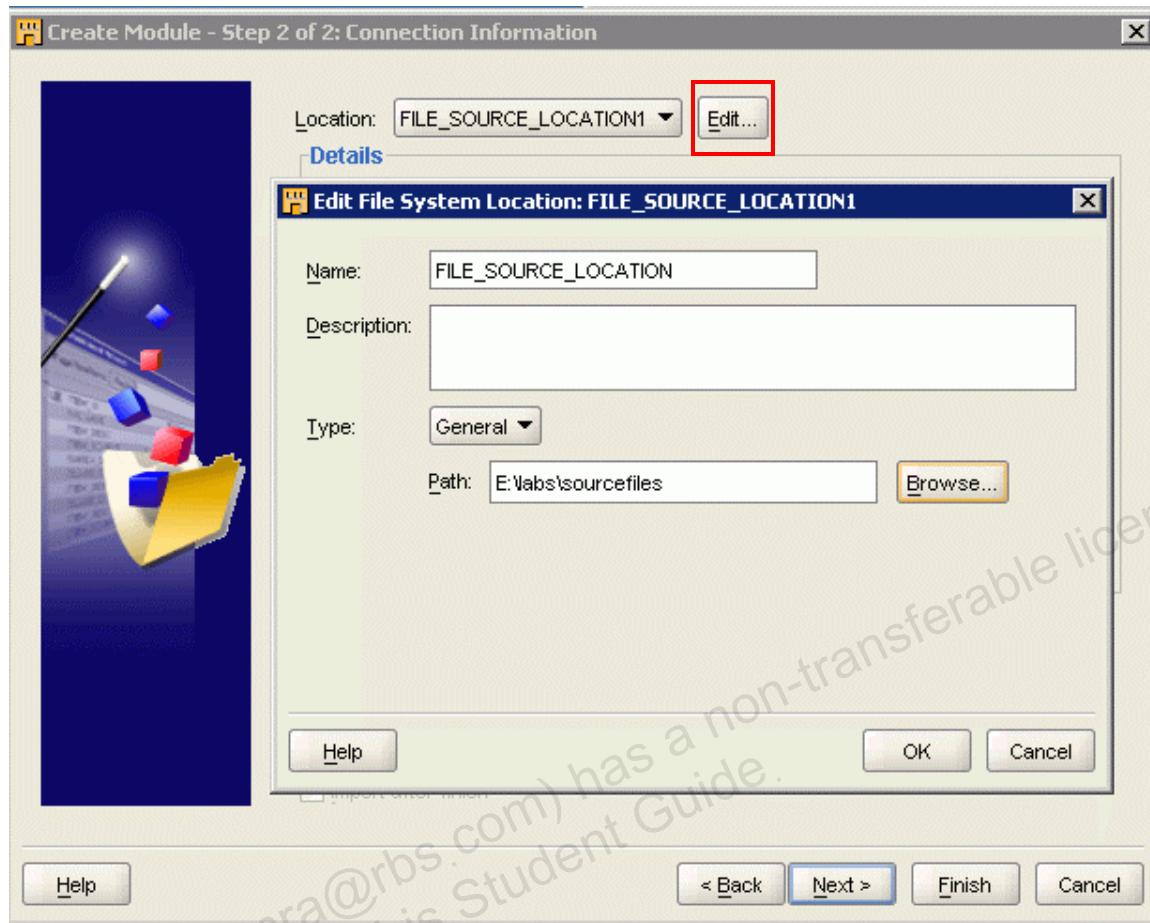
Click Next.

Solutions for Practice 3-1: Create a Flat File Source Module (continued)



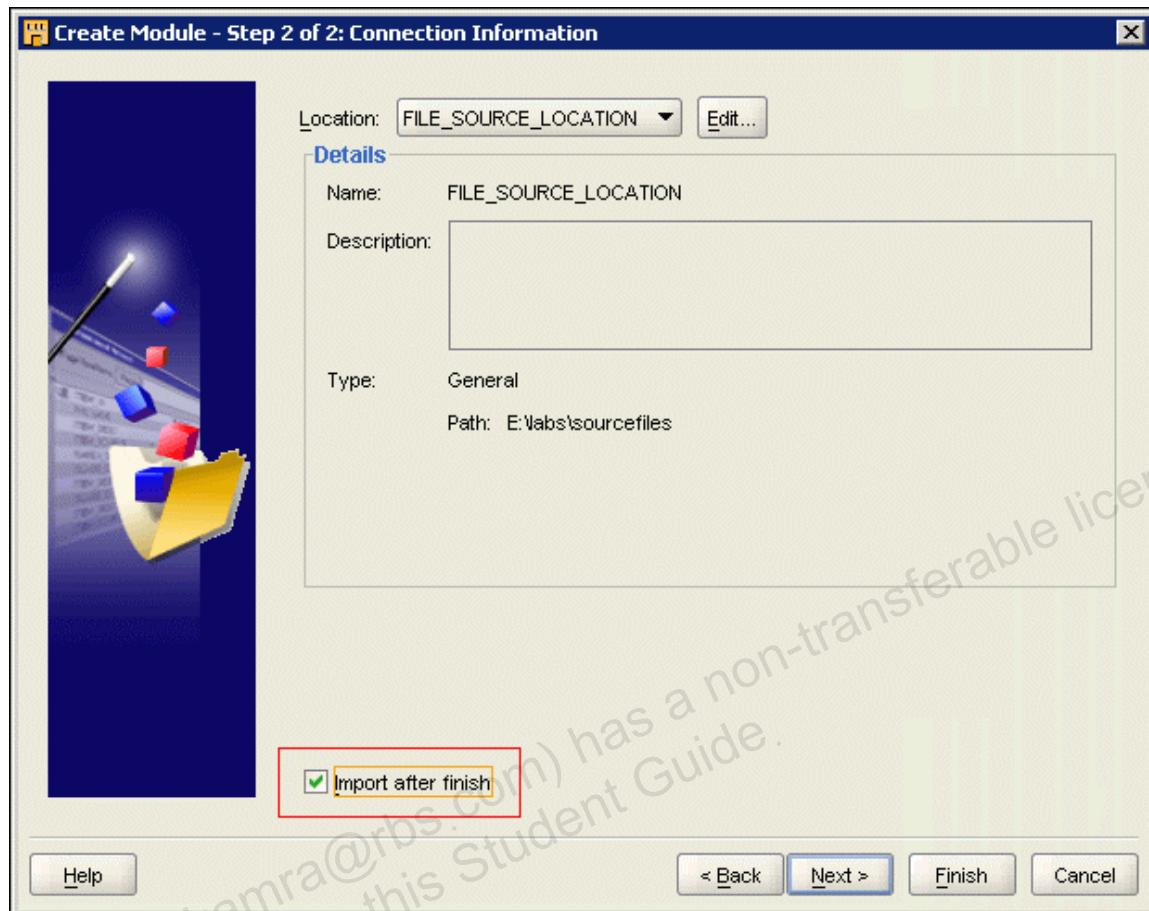
3. On the Connection Information page, click **Edit**. The Edit File System Locations window appears.

Solutions for Practice 3-1: Create a Flat File Source Module (continued)



4. In the Edit File System Locations window, change the location name from FILE_SOURCE_LOCATION1 to FILE_SOURCE_LOCATION. Click Browse to search for the drive and directory in the file system where the flat file resides. For this example, the flat file resides in **e:\labs\sourcefiles**. Click **OK**.
5. On the Connection Information page, select **Import after finish**, and click **Next**.

Solutions for Practice 3-1: Create a Flat File Source Module (continued)



6. Review the summary information, and click **Finish**. The FILE_SOURCE module is created, and the Import Metadata Wizard is launched.

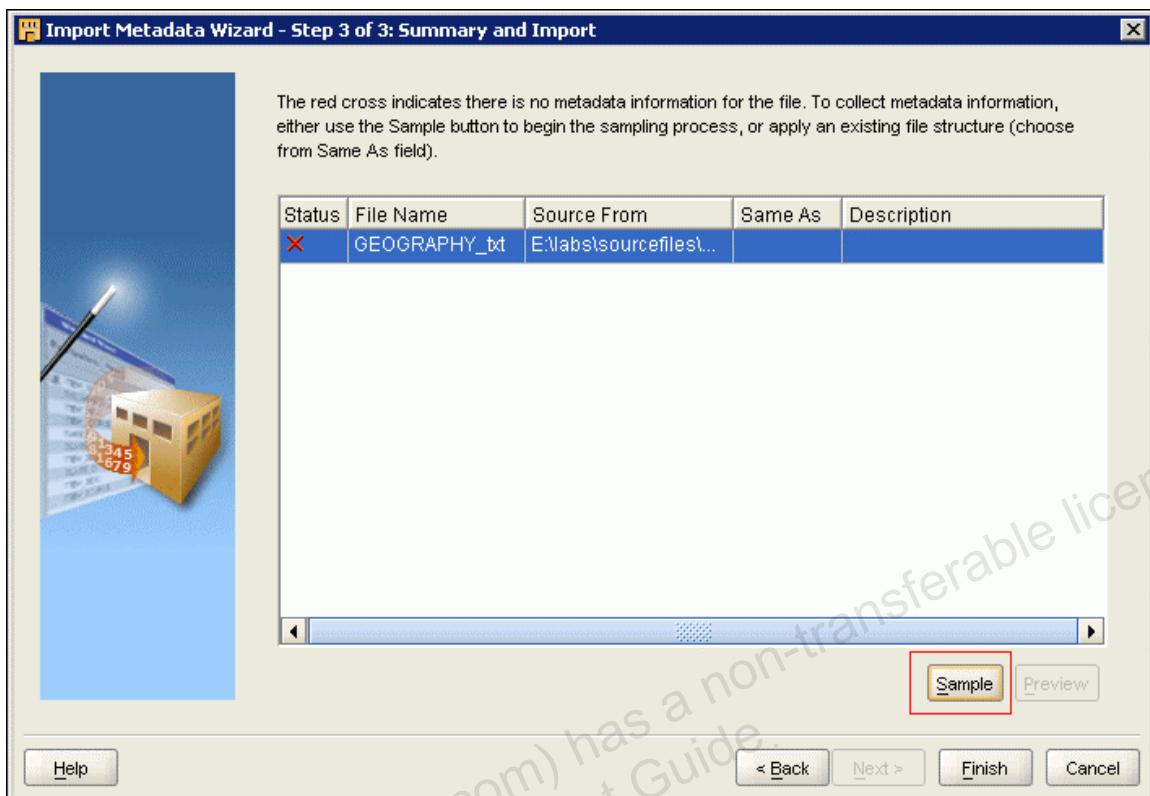
Import Flat File Metadata

- 2) Import the GEOGRAPHY.txt flat file.

Answer:

1. Click Next on the Welcome page of the Import Metadata Wizard.
2. On the Filter Information page, accept the default selection of **All Data Files**. Click **Next**.
3. On the Object Selection page, in the Available list, expand **e:\labs\sourcefiles** and select **GEOGRAPHY.txt**. Using the right-arrow button, move **GEOGRAPHY.txt** to the Selected list. Click **Next**.
4. On the Summary and Import page, the **GEOGRAPHY_txt** file is selected. Click **Sample**.

Solutions for Practice 3-1: Create a Flat File Source Module (continued)



Each time you use the Import Metadata Wizard to import metadata from existing flat files, the Import Metadata Wizard launches the Flat File Sample Wizard.

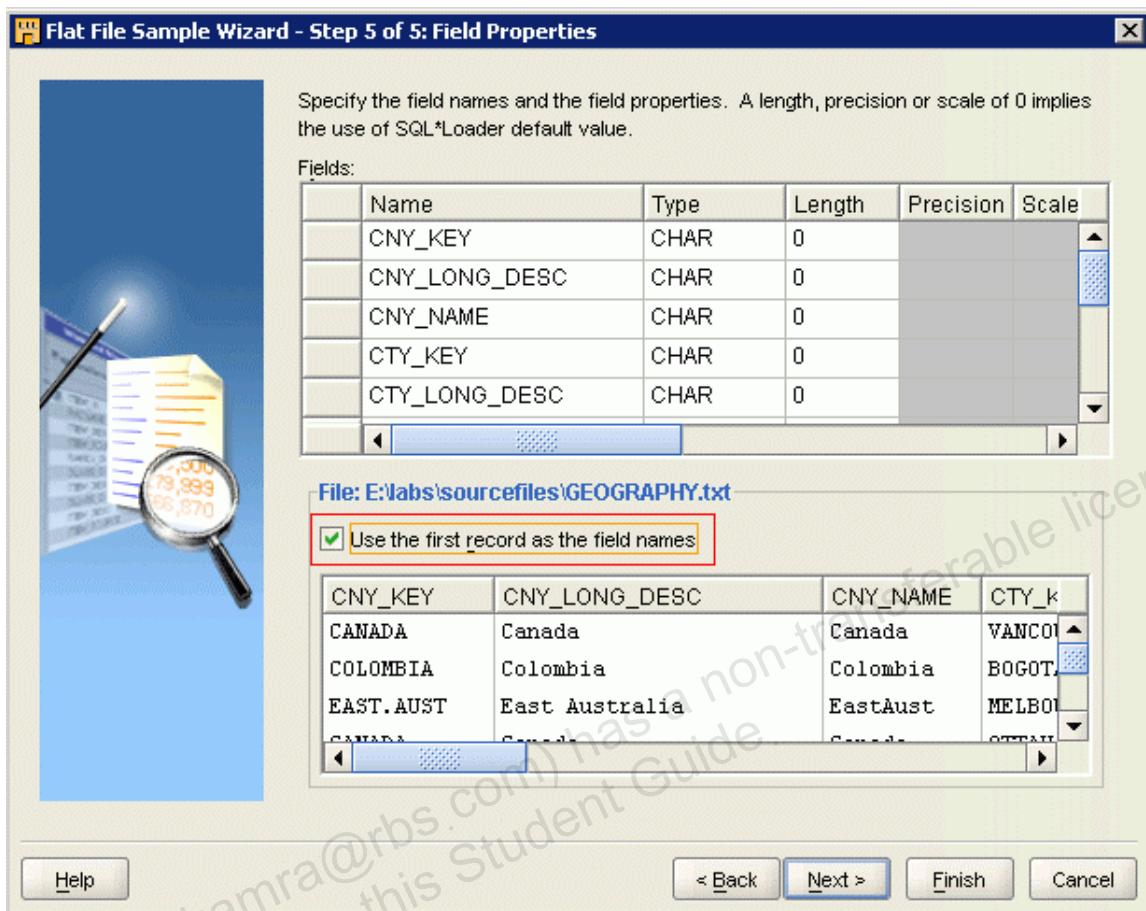
Sample the Flat File

- 3) Sample the **GEOGRAPHY.txt** file by using the Flat File Sample Wizard.

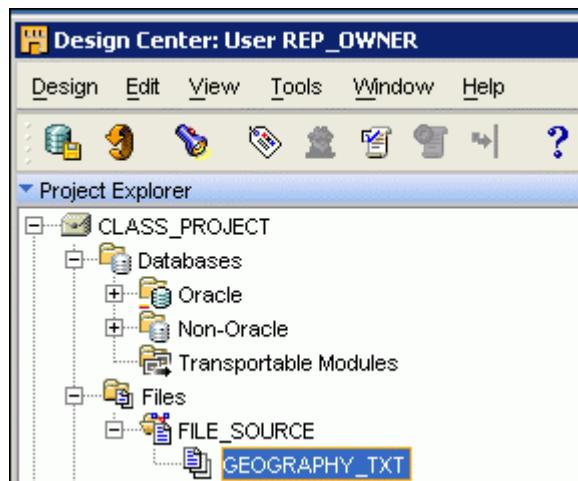
Answer:

1. Click **Next** on the Welcome page of the Flat File Sample Wizard.
2. On the Name page, accept the default name. Click **Next**.
3. On the Record Organization page, accept the default record delimiter. Click **Next**.
4. On the File Format page, also accept the default selection, and click **Next**.
5. On the File Layout page, accept the default selection, and click **Next**.
6. On the Field Properties page, select the **Use the first record as the field names** check box. Click **Next**.

Solutions for Practice 3-1: Create a Flat File Source Module (continued)



7. On the Summary page, click **Finish**.
8. Click Finish in the Import Metadata Wizard. The Import Progress dialog box displays the progress of the import. The dialog box closes when the import is successful. The flat file definition is complete.



- 4) Save your work.

Solutions for Practice 3-1: Create a Flat File Source Module (continued)



Answer:

1. In the Design Center, click Save All on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 3-2: Create a Relational Database Source Module

In this practice, you create a source module for relational database sources and import source object metadata into the source module.

Create a Relational Database Source Module

- 1) Create an Oracle source module with the following details:

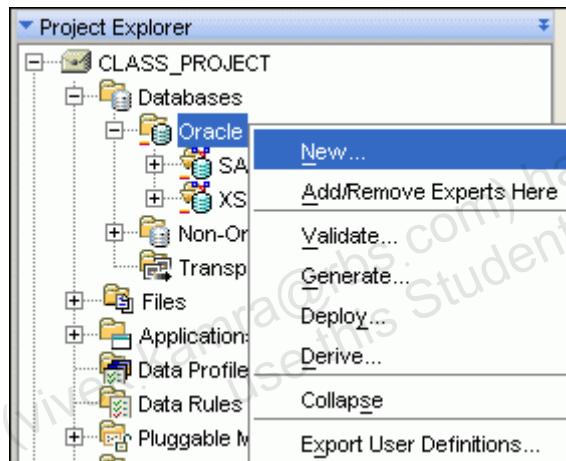
Source module name: HR

Description: This is an Oracle source module for the HR schema

Username/Password: HR/HR

Answer:

1. In the Project Explorer panel, expand CLASS_PROJECT > Databases. Right-click the Oracle node. Select New to open the Create Module Wizard. Click Next on the Welcome page.



2. On the Name and Description page, specify the following details:

Name: HR

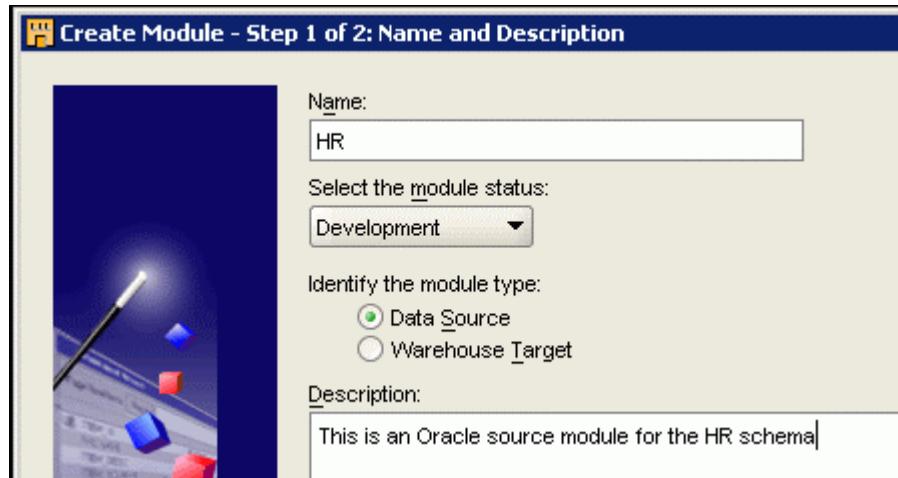
Select the module status: Development

Identify the module type: Data Source

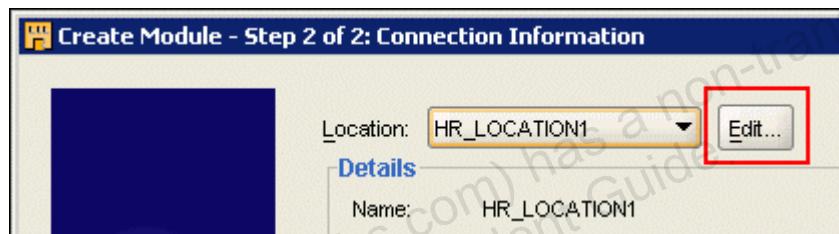
Description: This is an Oracle source module for the HR schema

Click Next.

**Solutions for Practice 3-2: Create a Relational Database Source Module
(continued)**



3. On the Connection Information page, click Edit. The Edit Oracle Database Location window appears.



4. In the Edit Oracle Database Location dialog box, specify the following information:

Name: change from HR_LOCATION1 to HR_LOCATION

User Name: HR

Password: HR

Host: localhost

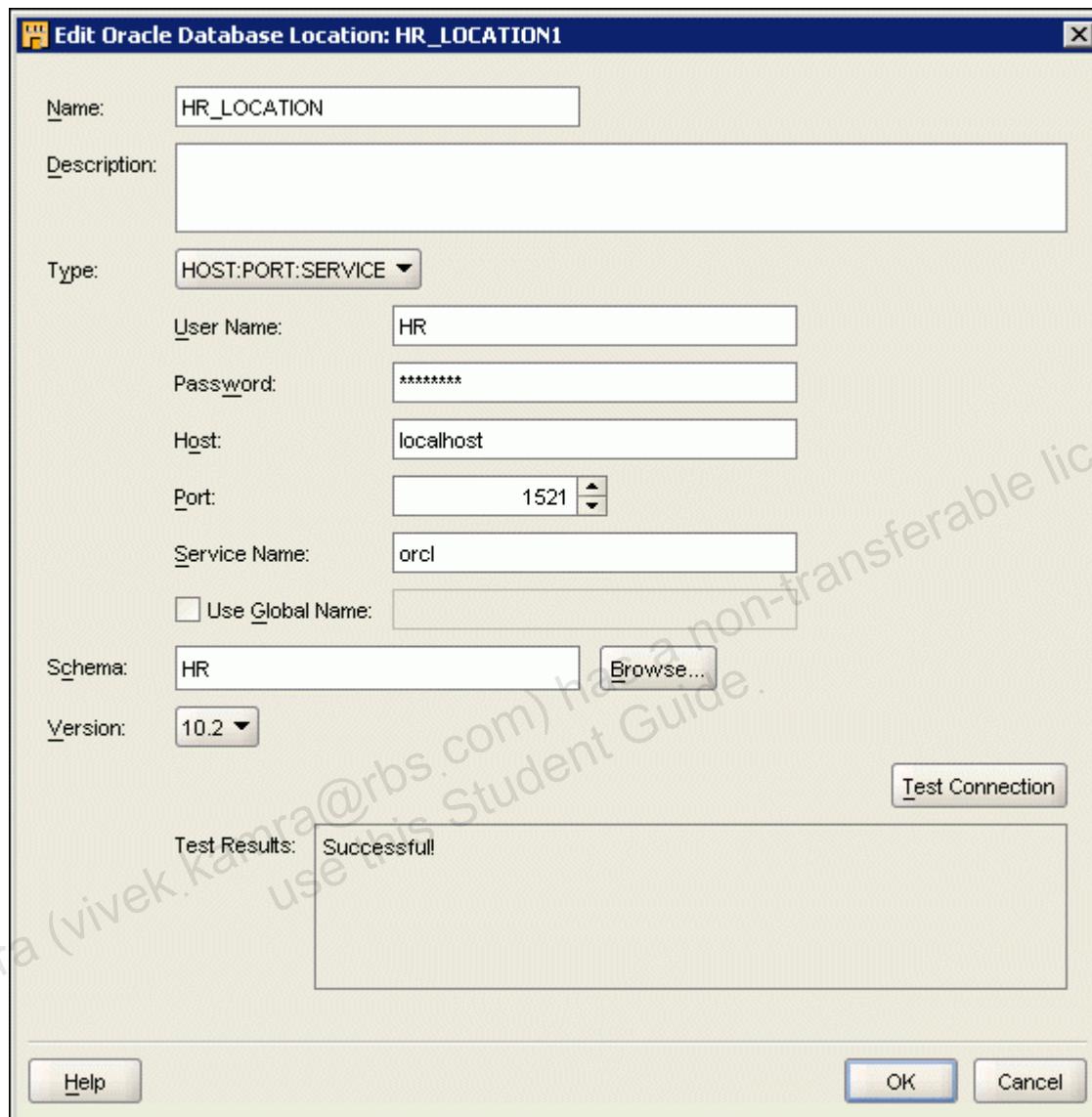
Port: 1521

Service Name: orcl

Version: 10.2

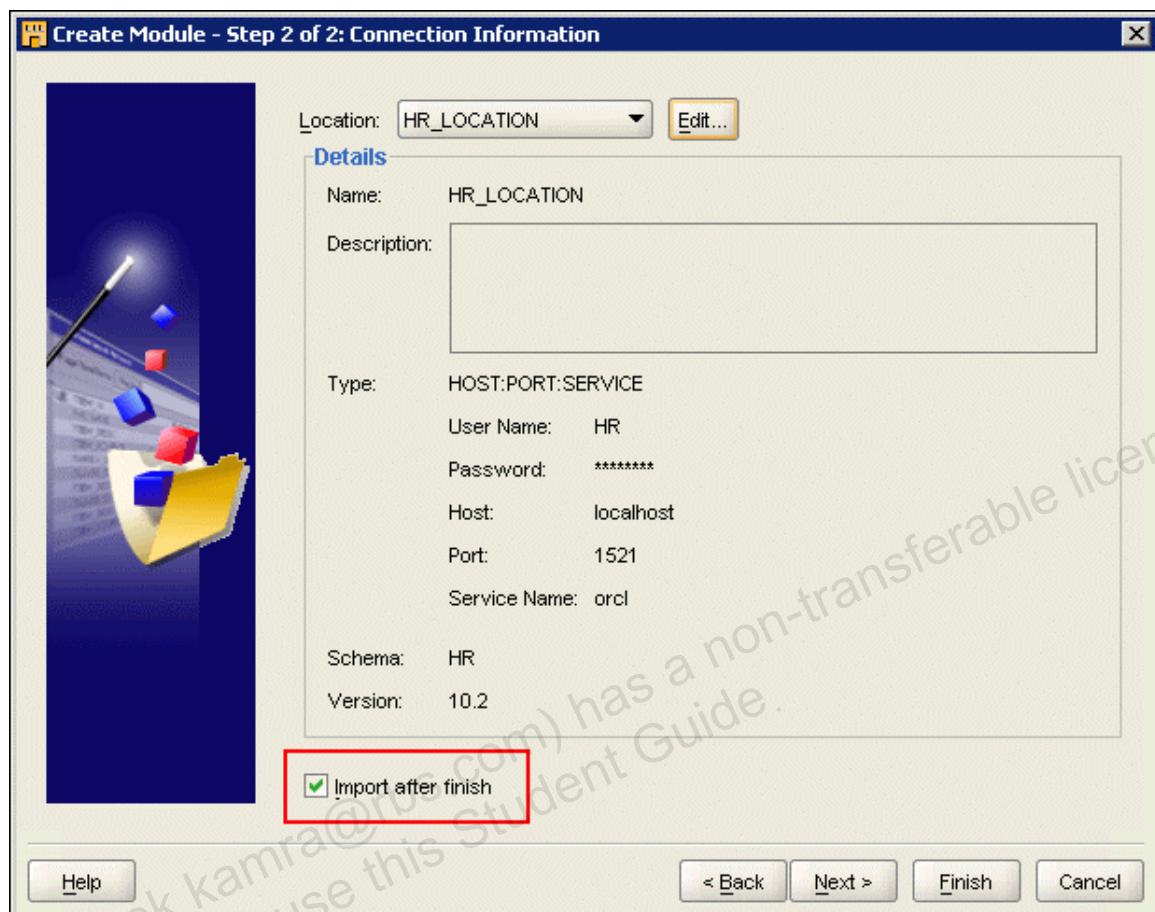
Click Test Connection. If the connection is successful, click OK.

**Solutions for Practice 3-2: Create a Relational Database Source Module
(continued)**



5. On the Connection Information page, ensure that **Import after finish** is selected. Click Next.

**Solutions for Practice 3-2: Create a Relational Database Source Module
(continued)**



6. On the Summary page, review the summary information, and click Finish.

The HR Oracle module is created, and the Import Metadata Wizard is launched.

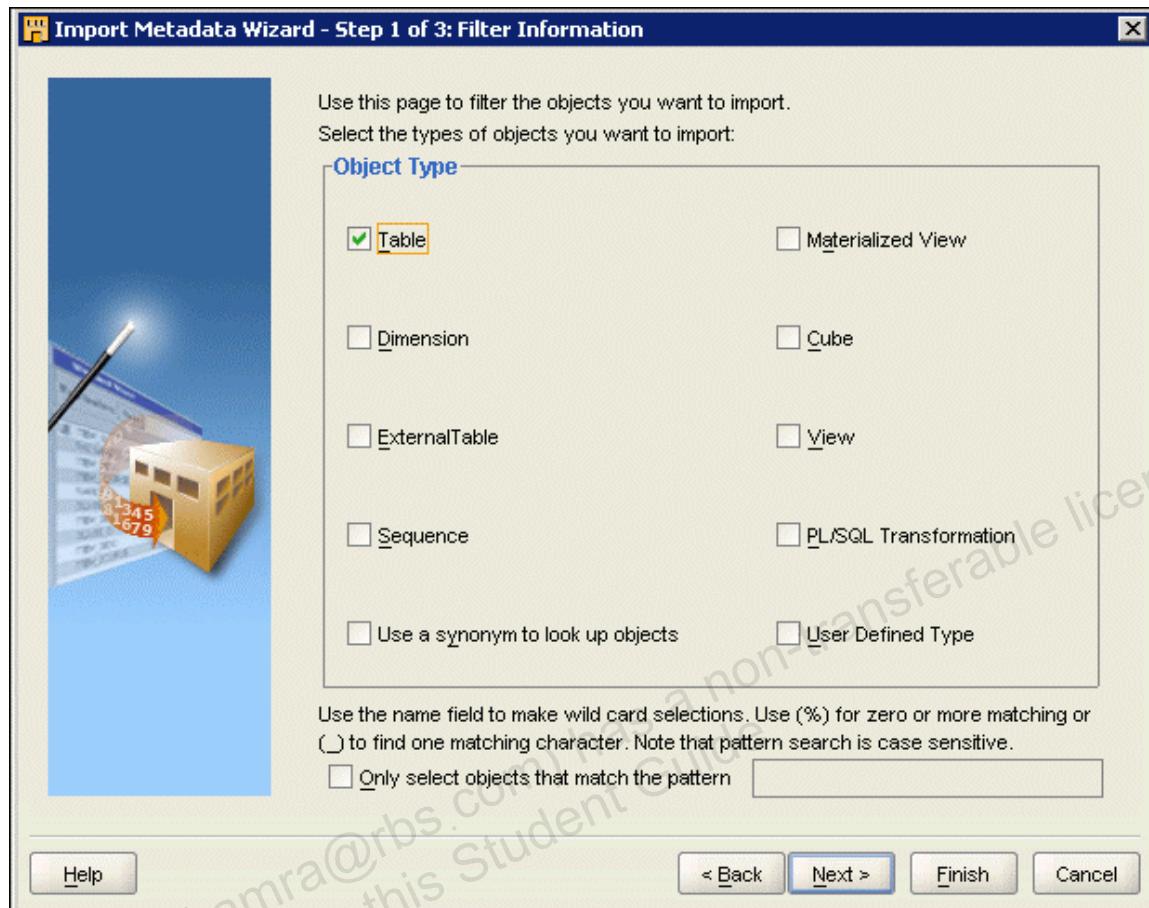
Import Metadata from the Relational Schema

- 2) Import all the tables from the HR schema.

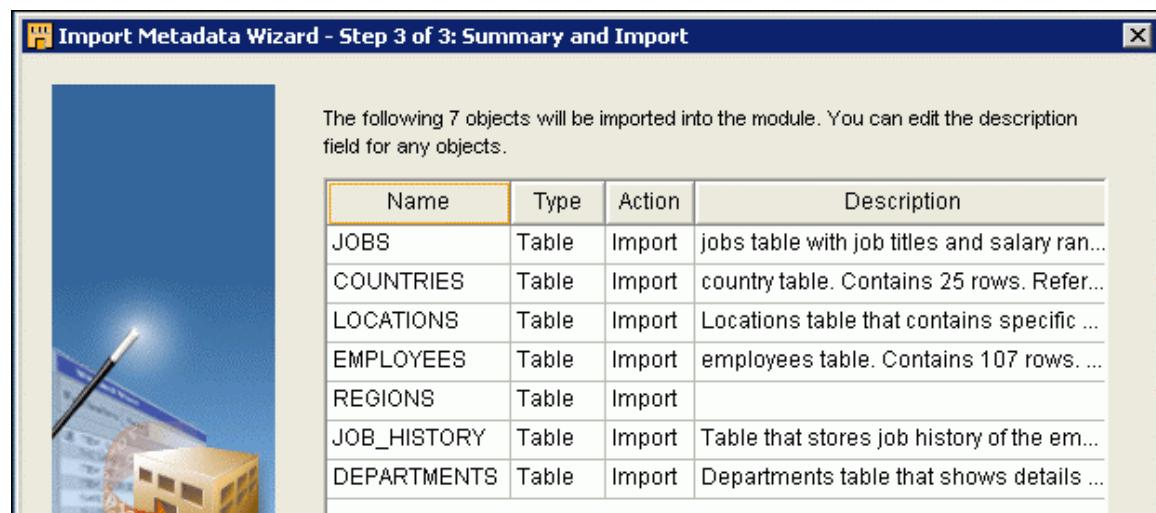
Answer:

1. Click Next on the Welcome page of the Import Metadata Wizard.
2. On the Filter Information page, select Tables, and deselect all other object types. Click **Next**.

Solutions for Practice 3-2: Create a Relational Database Source Module (continued)



3. On the Object Selection page, move the entire Table node from the Available list to the Selected list. Click Next.
4. On the Summary and Import page, confirm that you have all seven tables to be imported.



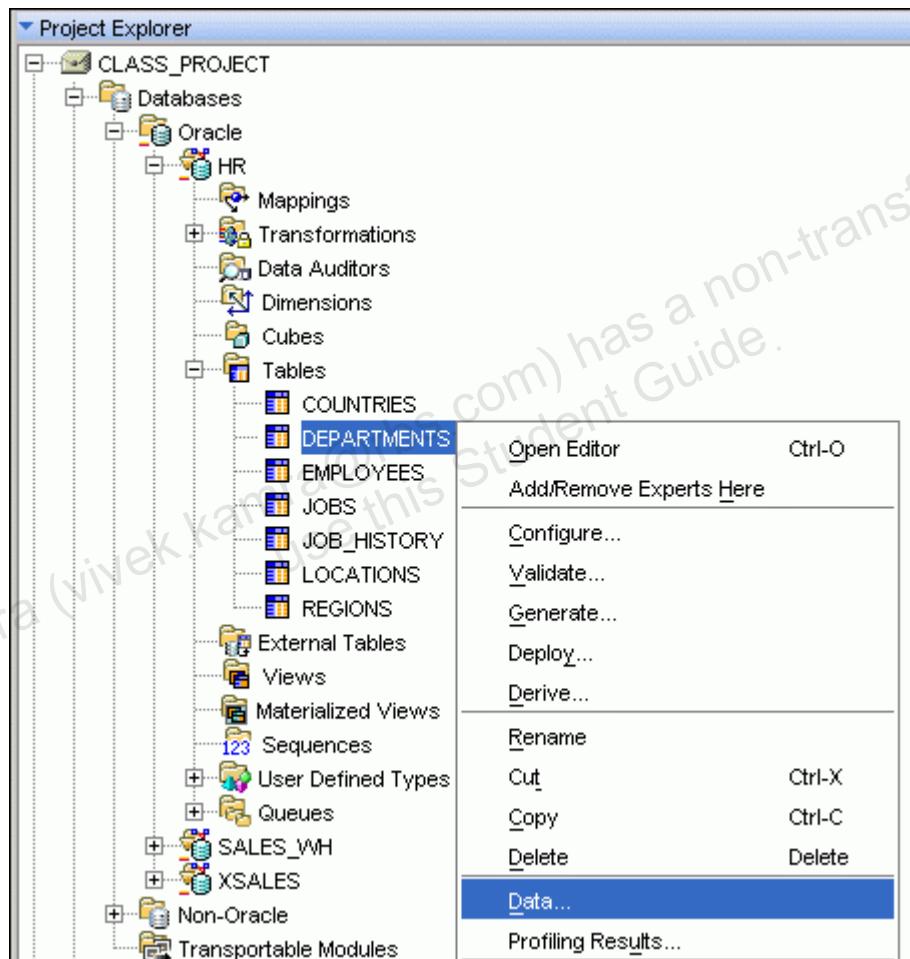
Solutions for Practice 3-2: Create a Relational Database Source Module (continued)

Click Finish. The Importing Progress dialog box displays the progress of the import. The Import Results dialog box displays the action against each object. Click OK to continue with the import of the objects. The import is now complete.

View Data in the HR Schema

Note: Here you are viewing data from the **HR source schema that already has data in the various tables in the schema**. OWB saves you the trouble of logging in to SQL*Plus and your having to know the syntax of SQL commands.

- 3) View the DEPARTMENTS table data.



Answer:

1. In the Project Explorer, expand CLASS_PROJECT > Databases > Oracle > HR > Tables.
2. Right-click DEPARTMENTS, and select Data. The Relational Data Viewer window appears with the DEPARTMENT data displayed.
- 4) Close the Relational Data Viewer.
- 5) Save your work.

**Solutions for Practice 3-2: Create a Relational Database Source Module
(continued)**

Answer:

In the Design Center, click the **Save All** icon on the toolbar. In the Warehouse Builder Warning dialog box, click **Yes**.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Practice Solutions for Lesson 4

In this practice, you create a data profile, and profile the data. This practice uses a slightly modified HR schema in your local database.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 4-1: Profile the Data

In this practice, you create a profile, and profile the data.

Before you begin creating the data profile, check whether you have access to source schema that you are going to profile. In this example, it is the HR schema.

- 1) Use the Connection Explorer to examine whether you have access to the HR schema.

Answer:

1. From the Connection Explorer panel, expand Locations > Databases > Oracle.
2. Double-click HR_LOCATION.
3. In the Edit Connection dialog box, all fields have values; enter HR in the password field (if not already entered), and click Test Connection. If the connection is successful click OK.

You now proceed to profile the data in the HR schema.

Create a Data Profile

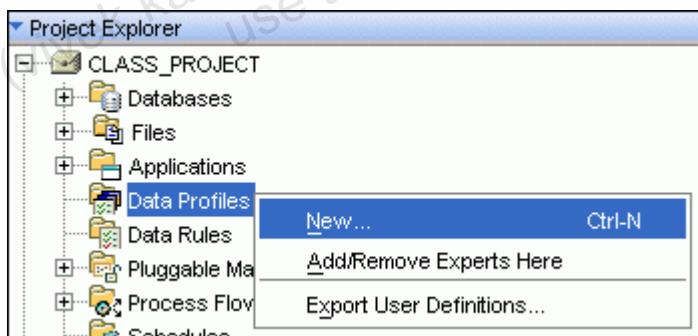
- 2) Create a Data Profile with the following details:

Name: DP_HR

Objects to be included: All tables from the HR schema

Answer:

1. In the Project Explorer, expand CLASS_PROJECT.
2. Right-click **Data Profiles**, and select New. The Create Data Profile Wizard is launched. Click Next on the Welcome page.



3. On the Name and Description page, enter the following information:

Name: **DP_HR**

Description: **Data Profiling the HR source data**

Click **Next**.

4. On the Select Objects page, select HR from the Available list, and move it to the Selected list. All tables from the HR schema will be listed in the Available list. Click **Next**.

Solutions for Practice 4-1: Profile the Data (continued)

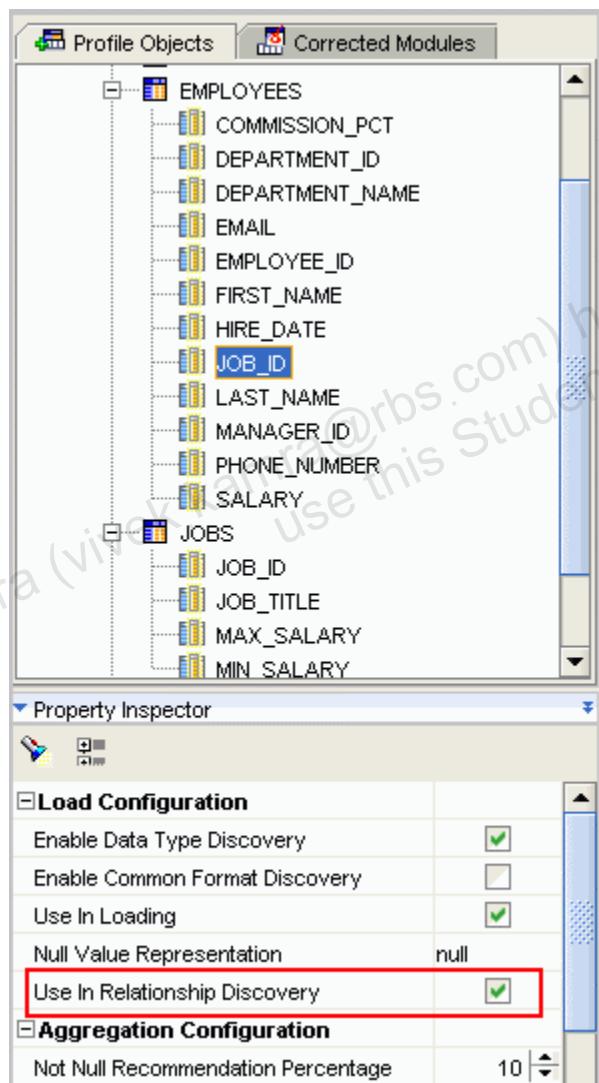
5. On the Summary page, review the settings. Click **Finish** to create the Data Profile. The DP_HR data profile is created, and the Data Profile Editor is launched.
6. Click OK in the Warehouse Builder Note dialog box.

Setting Configuration Parameters

- 3) Check whether the Use In Relationship Discovery property is selected for both JOBS.JOB_ID and EMPLOYEES.JOB_ID.

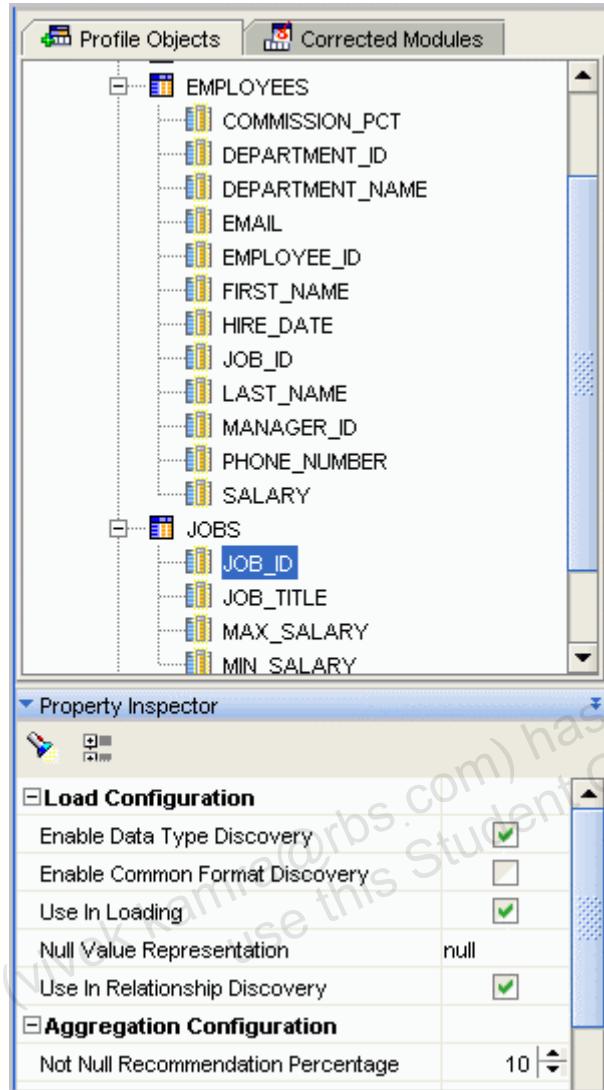
Answer:

1. Select EMPLOYEES.JOB_ID. The Use In Relationship Discovery property must be selected. (If not selected, select it now.)



5. Select JOBS.JOB_ID. The Use In Relationship Discovery property must be selected. (If not selected, select it now.)

Solutions for Practice 4-1: Profile the Data (continued)



- 4) Profile the data.
 - a) In the Data Profile Setup dialog box, enter the SYS user schema credentials as:

Username: sys

Password: oracle

- b) Details for the new profiling schema:

Username: profileworkspace

Password: profileworkspace

Note: You will be prompted to complete the Data Profile setup. The Data Profile setup is to create the data profiling workspace. The profiling workspace is a schema that OWB uses to store the results of the profiling job.

Answer:

1. In the Data Profile Editor, select **Profile > Profile** from the menu.

Solutions for Practice 4-1: Profile the Data (continued)

2. The Data Profile Setup dialog box is launched.

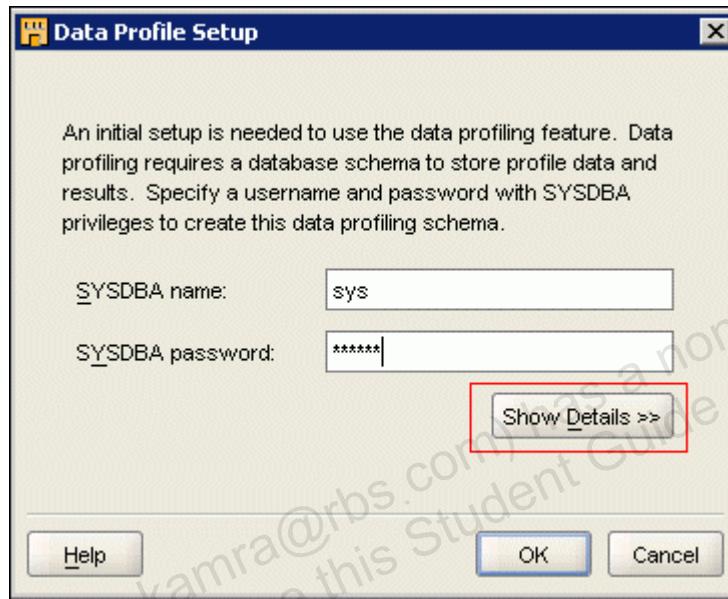
Note: This Data Profile Setup dialog box is a one-time step where you provide a name and password of the profiling schema that you want to create.

Enter the SYS user schema credentials as:

Username: sys

Password: oracle

Click **Show Details**.



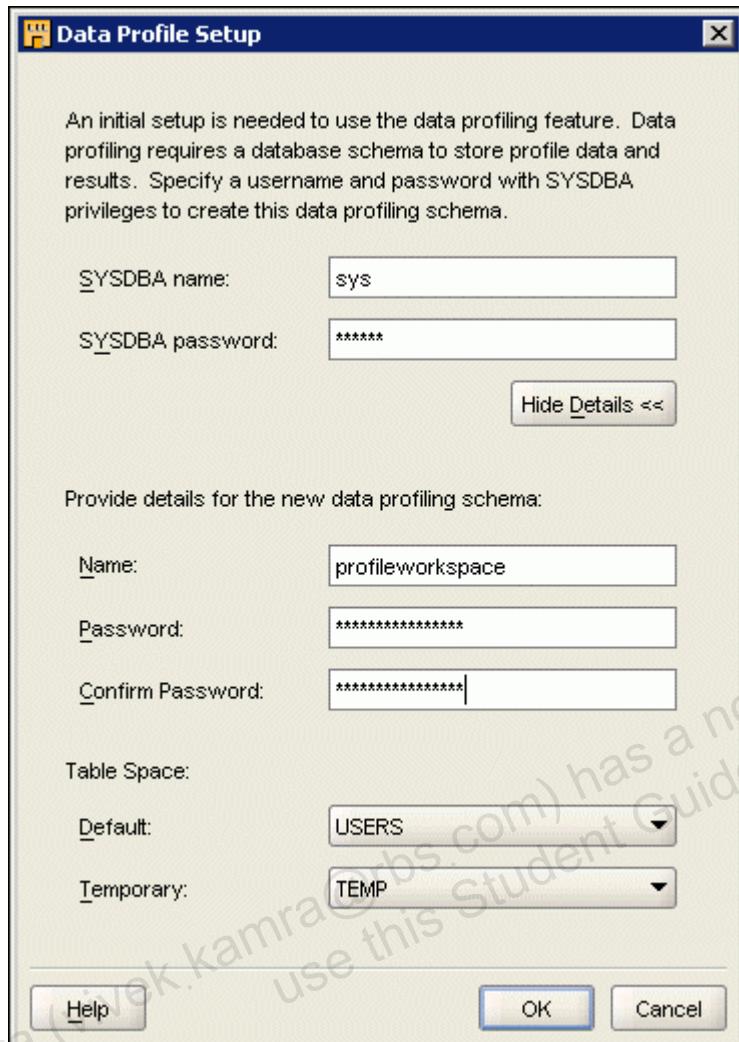
3. Enter the following details for the new profiling schema.

Note: Warehouse Builder uses default names and creates a database schema called *<name of the repository>_PRF* (for example, rep_owner_PRF) and a location called *<name of the repository>_location* (rep_owner_location).

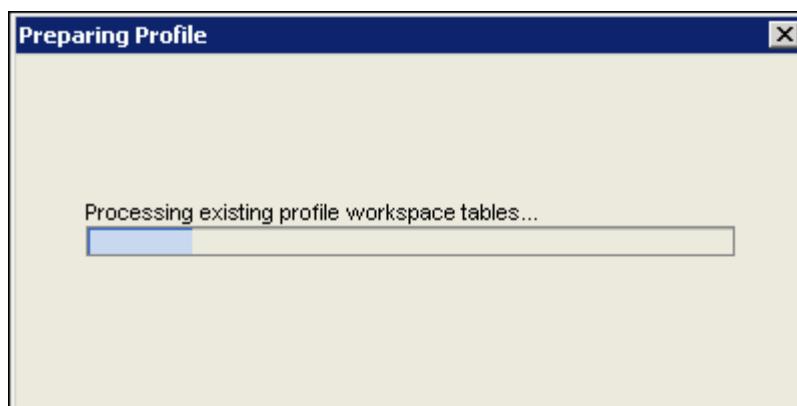
- a. Name: profileworkspace
- b. Password: profileworkspace
- c. Confirm Password: profileworkspace

Click OK.

Solutions for Practice 4-1: Profile the Data (continued)

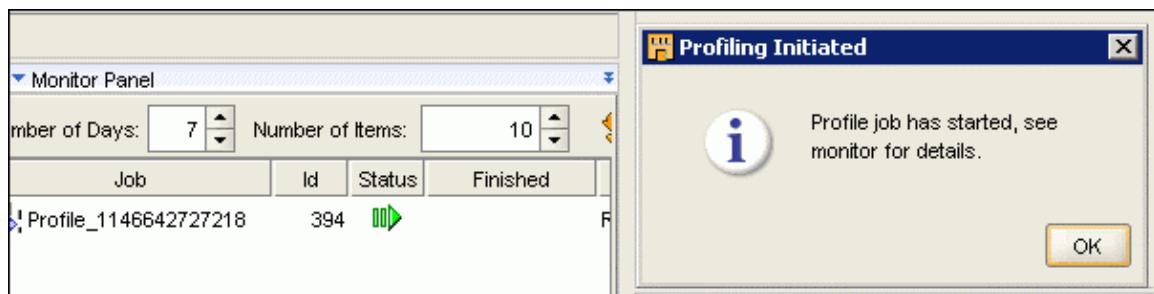


The Preparing Profile progress window appears. The window indicates the different objects that Warehouse Builder is creating in order to profile the data. (This may take 2–3 minutes.)

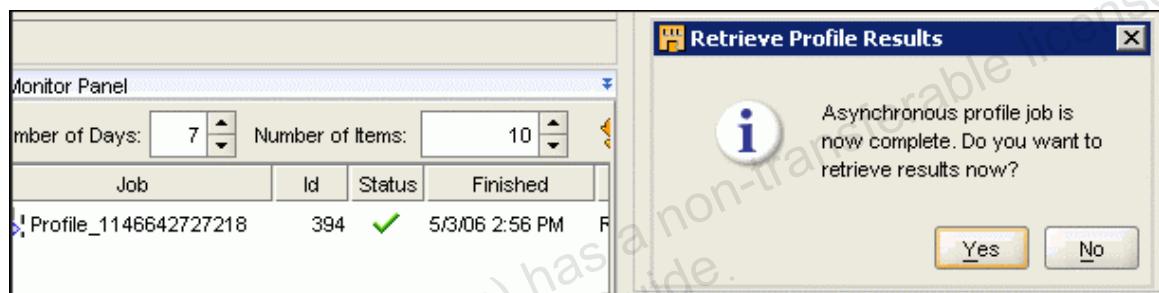


4. The Profiling Initiated dialog box appears. Click OK in the Profiling Initiated dialog box. From this point onward, you can see the status of your profiling operation in the Monitor Panel at the bottom-left corner of the Data Profile Editor.

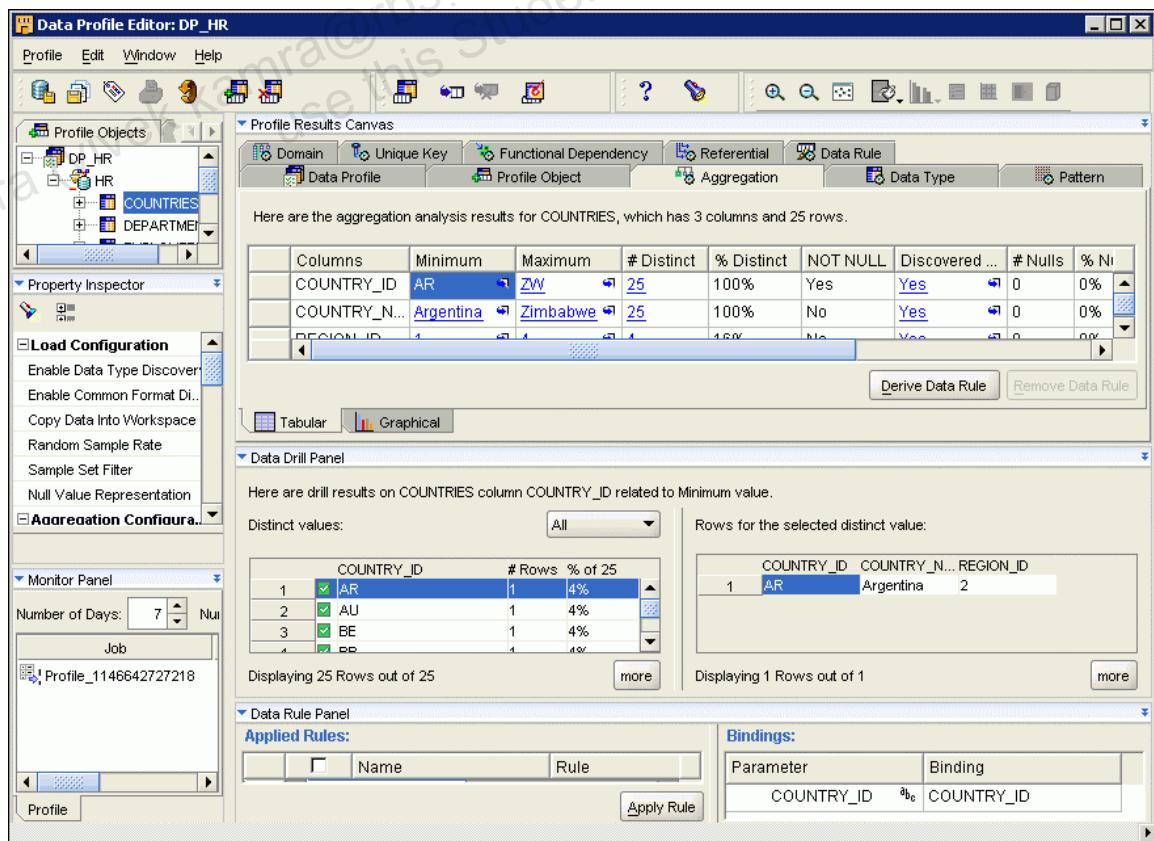
Solutions for Practice 4-1: Profile the Data (continued)



5. Profiling completes, and the Retrieve Profile Results dialog box appears. Click Yes to retrieve the profile results.



This completes the profiling actions, and you see the following window:



Solutions for Practice 4-1: Profile the Data (continued)

- 5) Save your work, and close the Data Profile Editor.

Answer:

1. In the Data Profile Editor, click Save All on the toolbar. Click **Yes** in the Warehouse Builder Warning dialog box.
2. From the **Profile** menu, select **Close**.

Solutions for Practice 4-2: View and Analyze the Profile Results

After having created the DP_HR data profile in Practice 4-1, you are now going to view the results and analyze the profile results.

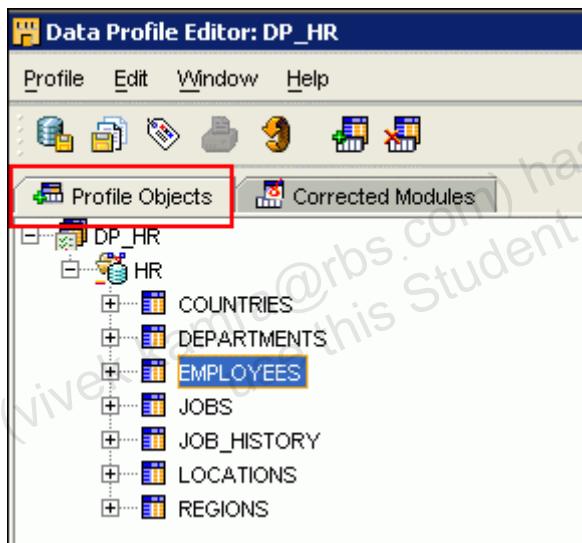
View the Profile Results

- 1) View the EMPLOYEES table data in the Profile Results Canvas.

Note: Open the Data Profile Editor.

Answer:

1. To open the DP_HR data profile:
 - a. In the Design Center, expand **CLASS_PROJECT > Data Profiles**, and double-click **DP_HR**.
2. To view the EMPLOYEES data:
 - a. In the Data Profile Editor, click the **Profile Objects** tab on the top left. Expand **DP_HR > HR**, and select **EMPLOYEES**.



- b. On the Profile Results Canvas, click the **Profile Object** tab.

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUM...	HIRE_DATE	JOB_ID	SALARY	COM
1	Donald	O'Connell	DOCONNEL	650.507.9833	21-JUN-99	SH_CLERK	2600	
2	Douglas	Grant	DGRANT	650.507.9844	13-JAN-00	SH_CLERK	2600	
3	Jennifer	Whalen	JWHALEN	515.123.4444	17-SEP-87	AD_ASST	4400	
4	Michael	Hartstein	MHARTSTE	515.123.5555	17-FEB-98	MK_MAN	13000	
5	Pat	Fay	PFAY	603.123.6666	17-AUG-97	MK_REP	6000	
6	Susan	Mavris	SMAVRIS	515.123.7777	07-JUN-94	HR_REP	6500	
7	Hermann	Baer	HBAER	515.123.8888	07-JUN-94	PR_REP	10000	
8	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-94	AC_MGR	12000	

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

After viewing the EMPLOYEES data, you now perform some exercises to see the attribute analysis power of OWB Data Profiling.

Attribute Analysis

- 2) You perform various kinds of attribute analysis with the EMPLOYEES table data.
 - a) List the maximum salary paid to an employee.

Answer:

1. Click the Aggregation tab on the Profile Results Canvas.
2. Scroll down to SALARY in the Columns column. In the same row, you see the maximum salary in the Maximum column.

Columns	Minimum	Maximum	# Distinct	% Distinct	NOT NULL	Discovered ...	# Nulls	% Nulls	Six-Sigma	Average
FIRST_NAME	Adam	Winston	91	83.5%	No	Yes	0	0%	7.00	
HIRE_DATE	17-JUN-... 1980	21-APR-... 1993	98	89.9%	Yes	Yes	0	0%	7.00	
JOB_ID	AC_ACCOUNT	ST_CLERK	17	15.6%	Yes	Yes	0	0%	7.00	
LAST_NAME	Abel	Zlotkey	102	93.6%	Yes	Yes	0	0%	7.00	
MANAGER_ID	100	205	18	16.5%	No	Yes	1	0.9%	3.86	125
PHONE_NUMBER	011.44.1... 555-1234	650.509.4... 555-1234	107	98.2%	No	Yes	0	0%	7.00	
SALARY	2100	24000	57	52.3%	No	Yes	0	0%	7.00	6463

- b) After finding the maximum salary value, find the details about the employee who earns this maximum salary.

Answer:

1. The maximum salary value is hyperlinked. Click the hyperlinked value to drill down into the data.
2. In the Data Drill panel on the right, you see the employee who earns that salary.

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

Profile Results Canvas

1. Data Rule tab selected.

Here are the aggregation analysis results for EMPLOYEES, which has 12 columns and 109 rows.

Columns	Minimum	Maximum	# Distinct	% Distinct	NOT NULL	Discovered ...	# Nulls	% Nulls	Six-Sigma	Average
FIRST_NAME	Adam	Winston	91	83.5%	No	Yes	0	0%	7.00	
HIRE_DATE	17-JUN-...	21-APR-...	98	89.9%	Yes	Yes	0	0%	7.00	
JOB_ID	AC_ACCOUNT	ST_CLERK	17	15.6%	Yes	Yes	0	0%	7.00	
LAST_NAME	Abel	Zlotkey	102	93.6%	Yes	Yes	0	0%	7.00	
MANAGER_ID	100	205	18	16.5%	No	Yes	1	0.9%	3.86	125
PHONE_NUMBER	011.44.1...	650.509....	107	98.2%	No	Yes	0	0%	7.00	
2. SALARY	2100	24000	57	52.3%	No	Yes	0	0%	7.00	6463

3. Row selection in the table.

4. Distinct values for SALARY.

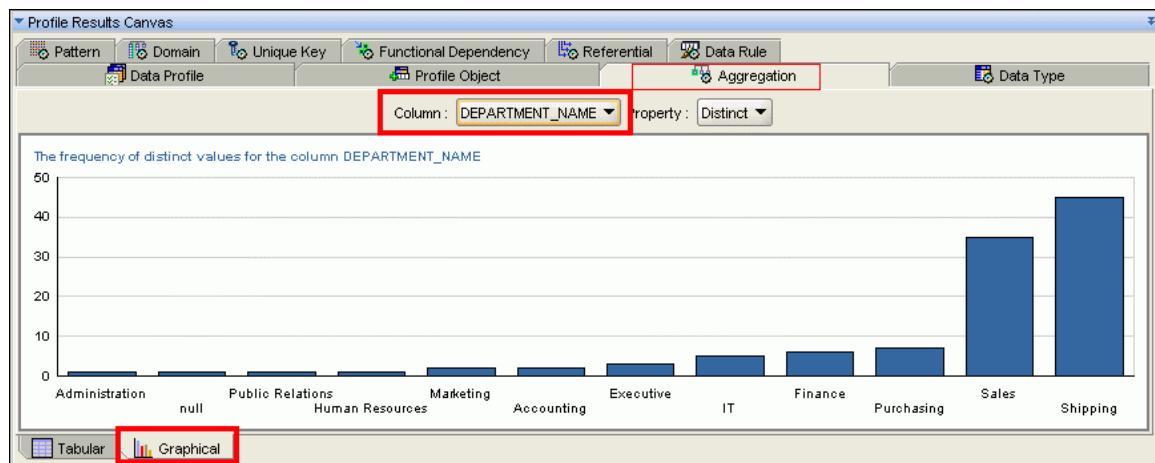
5. Rows for the selected distinct value: SALARY = 24000.

- c) Graphically display the distribution of employees across different departments.

Note: Use the Graphical subtab at the bottom of the Profile Results Canvas.

Answer:

1. On the Profile Results Canvas at the bottom, click the Graphical subtab.
2. From the Column drop-down list, select DEPARTMENT_NAME.

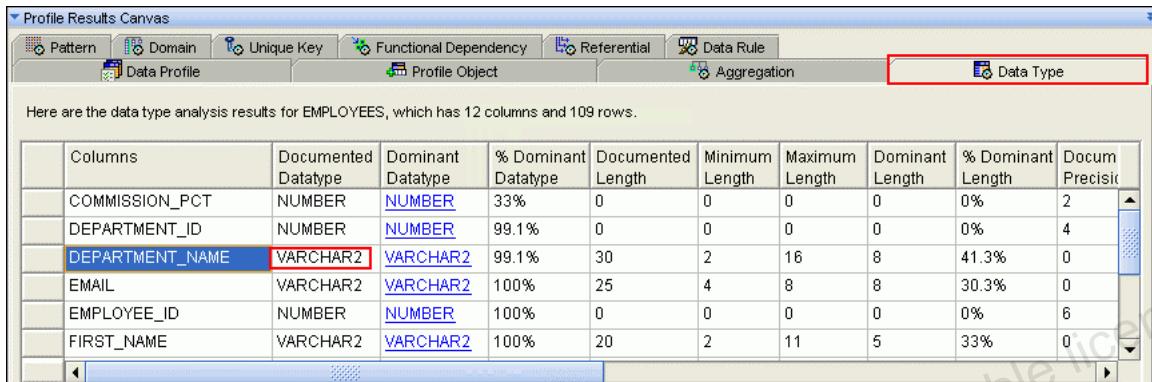


- d) Find out the data type for the DEPARTMENT_NAME column.

Answer:

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

1. On the Profile Results Canvas, click the Data Type tab. Click the Tabular tab if not already selected.
2. Scroll down to the DEPARTMENT_NAME row.
3. Check the value in the Documented Datatype column. It is VARCHAR2.



Columns	Documented Datatype	Dominant Datatype	% Dominant Datatype	Documented Length	Minimum Length	Maximum Length	Dominant Length	% Dominant Length	Docum Precisi
COMMISSION_PCT	NUMBER	NUMBER	33%	0	0	0	0	0%	2
DEPARTMENT_ID	NUMBER	NUMBER	99.1%	0	0	0	0	0%	4
DEPARTMENT_NAME	VARCHAR2	VARCHAR2	99.1%	30	2	16	8	41.3%	0
EMAIL	VARCHAR2	VARCHAR2	100%	25	4	8	8	30.3%	0
EMPLOYEE_ID	NUMBER	NUMBER	100%	0	0	0	0	0%	6
FIRST_NAME	VARCHAR2	VARCHAR2	100%	20	2	11	5	33%	0

- e) How many rows comply with the data type of the DEPARTMENT_NAME column?

Answer:

With the DEPARTMENT_NAME column selected, in the same row, scroll to the %Dominant Datatype column. It is 99.1%.

- f) Are there any departments that have not been assigned any department name? Display employee details that are not assigned to any department.

Answer:

1. In the DEPARTMENT_NAME row, click the hyperlinked value in the Dominant Datatype column.
2. Data is displayed in the Data Drill panel. In the Data Drill panel on the left, click the DEPARTMENT_NAME column, which is blank or null.

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

The screenshot shows the Data Profile Results Canvas with the 'Data Type' tab selected. A red box highlights the 'Data Type' tab in the top navigation bar. The main table displays data type analysis results for the EMPLOYEES table, which has 12 columns and 109 rows. The columns include COMMISSION_PCT, DEPARTMENT_ID, DEPARTMENT_NAME, EMAIL, EMPLOYEE_ID, and FIRST_NAME. The 'DEPARTMENT_NAME' column is highlighted with a red box. The table includes columns for Documented Datatype, Dominant Datatype, % Dominant Length, Documented Length, Minimum Length, Maximum Length, Dominant Length, % Dominant Length, and Document Precision.

Columns	Documented Datatype	Dominant Datatype	% Dominant Length	Documented Length	Minimum Length	Maximum Length	Dominant Length	% Dominant Length	Docum Precision
COMMISSION_PCT	NUMBER	NUMBER	33%	0	0	0	0	0%	2
2 DEPARTMENT_ID	NUMBER	NUMBER	99.1%	0	0	0	0	0%	4
3 DEPARTMENT_NAME	VARCHAR	VARCHAR2	99.1%	30	2	16	8	41.3%	0
EMAIL	VARCHAR2	VARCHAR2	100%	25	4	8	8	30.3%	0
EMPLOYEE_ID	NUMBER	NUMBER	100%	0	0	0	0	0%	6
FIRST_NAME	VARCHAR2	VARCHAR2	100%	20	2	11	5	33%	0

Data Drill Panel: Shows drill results on the DEPARTMENT_NAME column related to Datatypes. It lists distinct values (Administration, Accounting, Human Resources, Public Relations) and their counts (1, 2, 1, 1) and percentages (9%, 1.8%, .9%, .9%). A red box highlights the 'DEPARTMENT_NAME' column in the list. To the right, it shows rows for the selected distinct value 'Administration' with columns: COMMISSION_PCT, DEPARTMENT_ID, DEPARTMENT_NAME, EMAIL, EMPLOYEE_ID, FIRST_NAME. The row for 'Administration' is highlighted with a blue box.

- g) Find out the valid values for the JOB_ID column.

Answer:

1. On the Data Profile Results Canvas, click the Domain tab.
2. Scroll down to the JOB_ID column.
3. Examine its domain values in the Found Domain column.

The screenshot shows the Data Profile Results Canvas with the 'Domain' tab selected. A red box highlights the 'Domain' tab in the top navigation bar. The main table displays domain analysis results for the EMPLOYEES table, which has 12 columns and 109 rows. The columns include COMMISSION_PCT, DEPARTMENT_ID, DEPARTMENT_NAME, EMAIL, EMPLOYEE_ID, FIRST_NAME, HIRE_DATE, and JOB_ID. The 'JOB_ID' column is highlighted with a red box. The table includes columns for Columns, Found Domain, % Compliant, and Six-Sigma.

Columns	Found Domain	% Compliant	Six-Sigma
COMMISSION_PCT	.35 .1 .2 .3 .15 .25	32.1%	1.04
DEPARTMENT_ID	100 80 20 90 110 50 30 60	96.3%	3.29
DEPARTMENT_NAME	IT Executive Shipping Purchasing Finance Sales Market...	96.3%	3.29
EMAIL	.	0%	-6.25
EMPLOYEE_ID	.	0%	-6.25
FIRST_NAME	.	0%	-6.25
HIRE_DATE	.	0%	-6.25
JOB_ID	SH_CLERK ST_CLERK SA REP AD_VP PU_CLERK	89%	2.73

- h) Find the constraint name for the unique key constraint for the PHONE_NUMBER column.

Answer:

1. On the Data Profile Results Canvas, click the Unique Key tab.

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

2. In the Local Attribute(s) column, scroll down to the PHONE_NUMBER column.
In the Unique Key column, note that the constraint name is UK_9. This name may differ in your environment.

The screenshot shows the 'Profile Results Canvas' window. At the top, there are tabs: Pattern, Domain, Unique Key (which is highlighted with a red box), Functional Dependency, Referential, Data Rule, Data Profile, Profile Object, and Aggregation. Below the tabs, a message says: 'Here are the unique key analysis results for EMPLOYEES, which has 12 columns and 109 rows.' A table follows, with the last row (UK_9) highlighted with a blue box.

	Unique Key	Documented ?	Discovered ?	Local Attribute(s)	# Unique	% Uniq...	Six-Sigma
	EMP_EMAIL_UK	Yes	Yes	EMAIL	109	100%	7.00
	EMP_EMP_ID_PK	Yes	Yes	EMPLOYEE_ID	109	100%	7.00
	UK_6	No	Yes	FIRST_NAME	91	83.5%	2.47
	UK_7	No	Yes	HIRE_DATE	98	89.9%	2.78
	UK_8	No	Yes	LAST_NAME	102	93.6%	3.02
	UK_9	No	Yes	PHONE_NUMBER	107	98.2%	3.59

- i) Find whether there are any phone numbers that do not comply with the unique key constraint.

Answer:

Check the values in the %Unique column. If the value is not 100%, it means there are values in the PHONE_NUMBER column that are not unique.

- j) Display the phone number that does not comply with the unique key constraint.

Answer:

Click the hyperlinked Yes in the "Discovered?" column for PHONE_NUMBER. The phone numbers are displayed in the Data Drill Panel. In the Data Drill Panel, select Not Compliant from the "Distinct values" drop-down list. There are two phone numbers that are not unique.

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

Profile Results Canvas

Pattern Domain Unique Key Functional Dependency Referential Data Rule
Data Profile Profile Object Aggregation

Here are the unique key analysis results for EMPLOYEES, which has 12 columns and 109 rows.

	Unique Key	Documented ?	Discovered ?	Local Attribute(s)	# Unique	% Uniq...
	EMP_EMAIL_UK	Yes	Yes	EMAIL	109	100%
	EMP_EMP_ID_PK	Yes	Yes	EMPLOYEE_ID	109	100%
	UK_6	No	Yes	FIRST_NAME	91	83.5%
	UK_7	No	Yes	HIRE_DATE	98	89.9%
	UK_8	No	Yes	LAST_NAME	102	93.6%
	UK_9	No	Yes	PHONE_NUMBER	107	98.2%

Tabular Graphical Joins

Data Drill Panel

Here are drill results on EMPLOYEES column PHONE_NUMBER related to Unique Key.

Distinct values: Not Compliant Rows for the selected distinct value:

	PHONE_NUMBER	# Rows	% of 109
1	011.44.1344.129268	2	1.8%
2	515.127.4562	2	1.8%

- k) Find the employees who share the same number.

Answer:

In the Data Drill Panel, click the phone number value. In the Data Drill panel on the right, the employees who share the same phone number are displayed.

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

The screenshot shows the Oracle Data Profiler interface. The top navigation bar includes tabs for Pattern, Domain, Unique Key (which is selected and highlighted in red), Functional Dependency, Referential, Data Rule, Data Profile, Profile Object, Aggregation, and Data Type. Below the tabs, a message states: "Here are the unique key analysis results for EMPLOYEES, which has 12 columns and 109 rows." A table displays the analysis results:

	Unique Key	Documented ?	Discovered ?	Local Attribute(s)	# Unique	% Uniq...	Six-Sigma	
	EMP_EMAIL_UK	Yes	Yes	EMAIL	109	100%	7.00	
	EMP_EMP_ID_...	Yes	Yes	EMPLOYEE_ID	109	100%	7.00	
	UK_6	No	Yes	FIRST_NAME	91	83.5%	2.47	
	UK_7	No	Yes	HIRE_DATE	98	89.9%	2.78	
	UK_8	No	Yes	LAST_NAME	102	93.6%	3.02	
	UK_9	No	Yes	PHONE_NUMBER	107	98.2%	3.59	

Buttons at the bottom right include "Derive Data Rule" and "Remove". Below the table are tabs for Tabular, Graphical, and Joins. A "Data Drill Panel" section shows drill results for the PHONE_NUMBER column, indicating two distinct values: 011.44.1344.129268 and 515.127.4562, each with 1.8% of the rows.

Now you work with the next type of Data Profiling, that is, Referential Analysis. You see existing foreign keys that were documented in the data dictionary, as well as relationships found by data profiling.

3) Perform Referential Analysis

- Note down the relationship name and type for the EMPLOYEES.JOB_ID column.

Answer:

On the Profile Results Canvas, click the Referential tab. In the Local Attribute column, scroll down to the JOB_ID attribute. In the same row of the Relationship column, note down the relationship name. (For this example, it is FK_17. The relationship name may be different in your environment).

The screenshot shows the Oracle Data Profiler interface with the Referential tab selected. The left sidebar shows a tree view of profile objects, including DP_HR, HR, COUNTRIES, DEPARTMENTS, EMPLOYEES, JOBS, JOB_HISTORY, LOCATIONS, and REGIONS. The right panel displays referential analysis results for the EMPLOYEES table:

Relationship	Type	...	Discovered ?	Local Attribute(s)	Remote Key	Remote Attribute(...)	Remote Relation
RR_17	Row Relationship	No	Yes	DEPARTMENT_ID	RR_8	DEPARTMENT_ID	JOB_HISTORY
EMP_DEPT_FK	Foreign Key	...	Yes	DEPARTMENT_ID	DEPT_ID_PK	DEPARTMENT_ID	DEPARTMENTS
EMP_MANAGER...	Foreign Key	...	Yes	MANAGER_ID	EMP_EMP_ID...	EMPLOYEE_ID	EMPLOYEES
FK_17	Foreign Key	No	Yes	JOB_ID	UK_17	JOB_ID	JOB

- Find out the percentage of records that comply with this foreign key relationship.

Answer:

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

In the same row, navigate to the %Compliant column. The value is 98.2%.

Profile Results Canvas

Referential tab selected.

Here are the referential analysis results for EMPLOYEES, which has 12 columns and 109 rows.

	Relationship	Type	...	Local Attribute(s)	% Compliant	...
	RR_17	Row Relationship	No	DEPARTMENT_ID	...	J...	16	85.3%	...
	EMP_DEPT_FK	Foreign Key	...	DEPARTMENT_ID	0	100%	...
	EMP_MANAGE...	Foreign Key	...	MANAGER_ID	0	100%	...
	FK_17	Foreign Key	No	JOB_ID	...	J...	J...	...	2	98.2%	...

- c) Which JOB_IDs did not comply with the foreign key relationship?

Answer:

In the same row, click the hyperlinked Yes in the “Discovered?” column to drill down on the data. In the Data Drill Panel, select Not Compliant from the “Distinct values” drop-down list.

Profile Results Canvas

Referential tab selected.

Here are the referential analysis results for EMPLOYEES, which has 12 columns and 109 rows.

	Relationship	Type	...	Discovered ?	Local Attribute(s)
	RR_17	Row Relationship	No	Yes	DEPARTMENT_ID	...	J...
	EMP_DEPT_FK	Foreign Key	...	Yes	DEPARTMENT_ID
	EMP_MANAGE...	Foreign Key	...	Yes	MANAGER_ID
	FK_17	Foreign Key	No	Yes	JOB_ID	...	J...	J...

Tabular, Graphical, Joins, Redundant Columns tabs available.

Data Drill Panel

Here are drill results on EMPLOYEES column JOB_ID related to Foreign Key.

Distinct values: Not Compliant

JOB_ID	# Rows	% of 109
1 PU_CLIRK	1	.9%
2 SA_REPP	1	.9%

Rows for the selected distinct value:

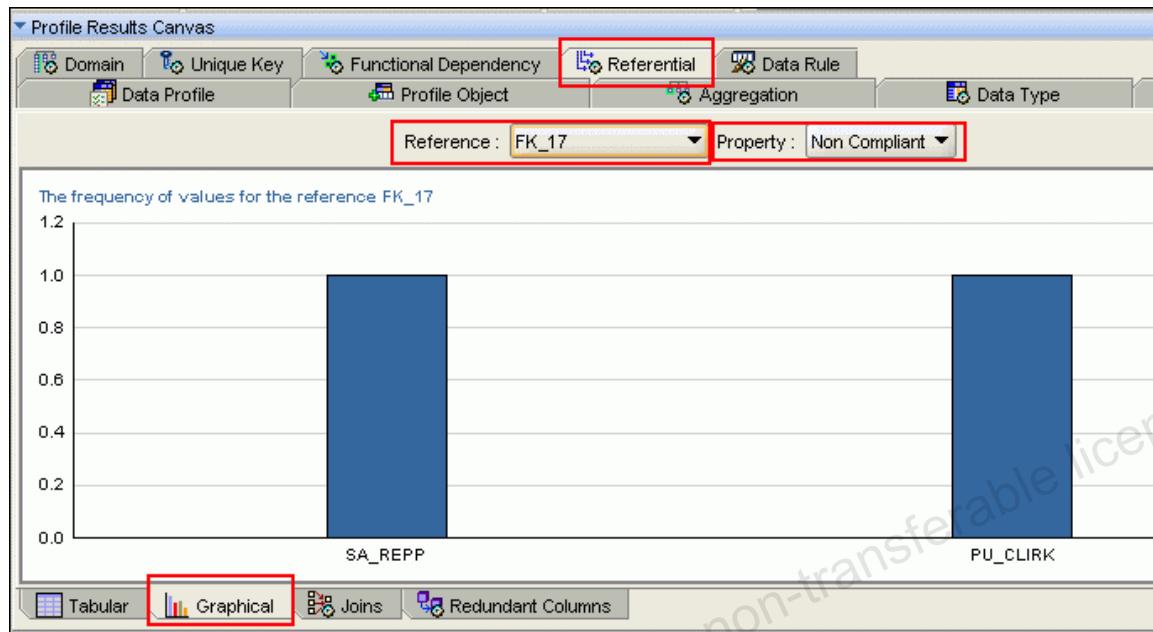
- d) View the noncompliant records in a graphical view.

Answer:

With the Referential tabbed page displayed on the Profile Results Canvas, click the Graphical subtab at the bottom of this canvas. From the Reference drop-down list,

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

select the relationship name that you noted in step 1 (in this example, FK_17). From the Property drop-down list, select Non Compliant.

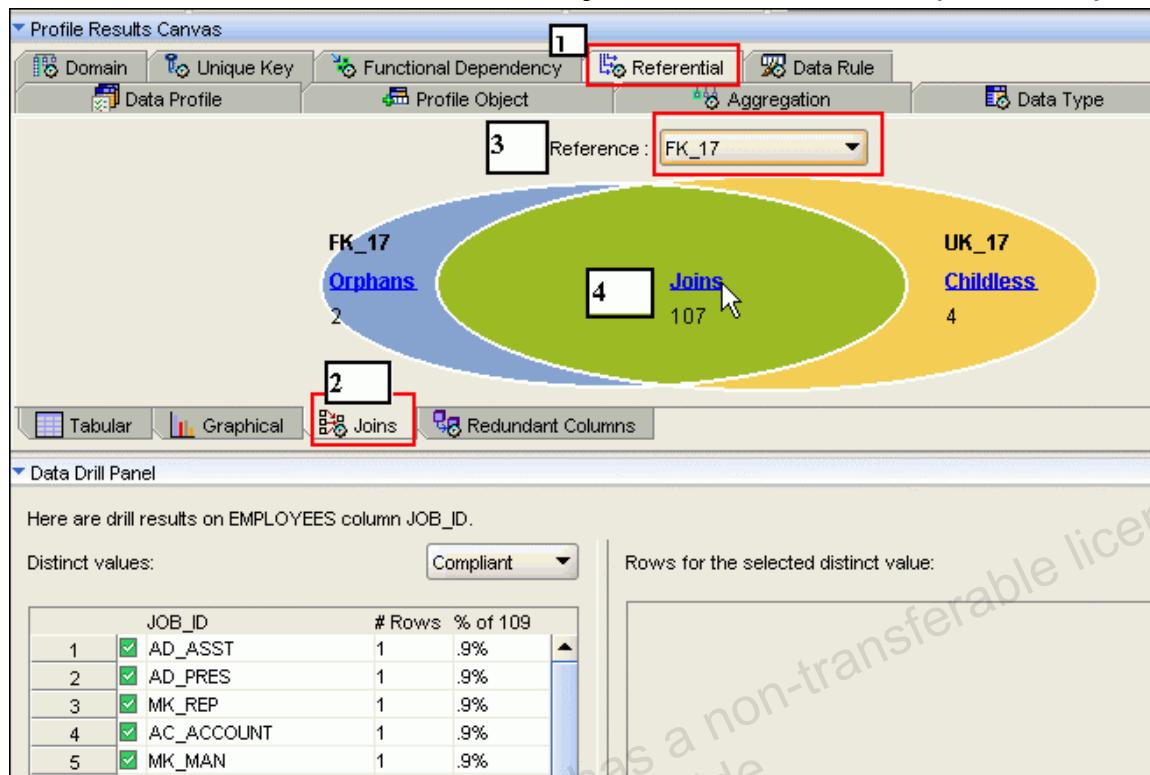


- e) Display the JOB_IDs found in both the EMPLOYEES and JOBS tables.

Answer:

Click the Joins tab on the Referential tabbed page. From the Reference drop-down list, select the relationship name that you noted in step 1. (In this example, it is FK_17). Click Joins. In the Data Drill Panel, you see a listing of the JOB_IDs that were found in both the tables.

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)



- f) Display the JOB_IDs found in the EMPLOYEES table, but not found in the JOBS table (also called orphans).

Answer:

On the Joins tabbed page, ensure that the relationship name that you noted in step 1 is selected (for this example, FK_17). Click the Orphans hyperlink. The result is that two JOB_IDs are found in the EMPLOYEES table, but not in the JOBS table.

- g) Display the JOB_IDs found in the JOBS table but not found in the EMPLOYEES table (also called childless).

Answer:

On the Joins tabbed page, ensure that the relationship name that you noted in step 1 is selected (for this example, FK_17). Click Childless. There are four JOB_IDs that are found in the JOBS table but not in the EMPLOYEES table.

Another type of data profiling is the functional dependency. The Functional Dependency tabbed page displays information about the attribute or attributes that seem to depend or determine other attributes.

Functional Dependencies Analysis

- 4) Find out which attributes are dependent, and on which attribute they depend.

Answer:

Solutions for Practice 4-2: View and Analyze the Profile Results (continued)

On the Profile Results Canvas, click the Functional Dependency tab. Ensure EMPLOYEES is selected in the left panel. From the Show drop-down list, select **Only 100% dependencies**. DEPARTMENT_NAME is dependent on DEPARTMENT_ID.

The screenshot shows the Oracle Warehouse Builder interface. On the left, the 'Profile Objects' pane displays a tree structure with 'DP_HR' expanded, showing 'HR' which further expands into 'COUNTRIES', 'DEPARTMENTS', 'EMPLOYEES' (selected), 'JOBS', 'JOB_HISTORY', 'LOCATIONS', and 'REGIONS'. The main area is titled 'Profile Results Canvas' and contains tabs for 'Domain', 'Unique Key', 'Functional Dependency' (which is highlighted with a red box), 'Referential', 'Data Rule', and 'Aggregation'. A sub-tab 'Data Profile' is also visible. Below the tabs, a message states: 'Here are the functional dependency analysis results for EMPLOYEES, which has 12 columns and 109 rows.' A dropdown menu labeled 'Show:' is set to 'Only 100% dependencies'. A table below lists one dependency:

Determinant	Dependent	# Defects	% Compliant
DEPARTMENT_ID	DEPARTMENT_NAME	0	100%

- 5) Close the Data Profile Editor.

Answer:

In the Data Profile Editor, select Close from the Profile menu.

- 6) Save your work.

Answer:

In the Design Center, click the Save All icon on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

Practice Solutions for Lesson 5

In this practice, you create a target user that is your actual data warehouse schema. You then create the staging warehouse module and an external table.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 5-1: Create an External Table

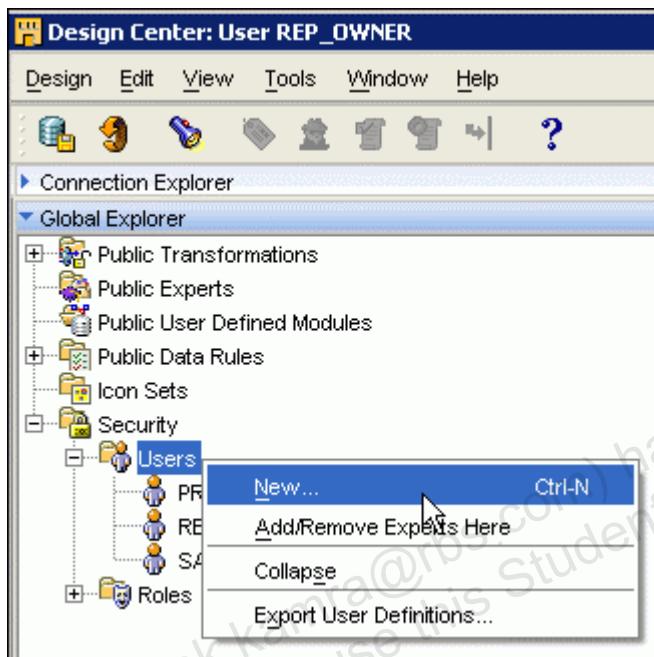
In this practice, before you create an external table, you create a target user, a target module that acts as a staging module, and finally the external table.

Create a Target User

- 1) Create a target user with the following details:

Name: **STAGING_USER**

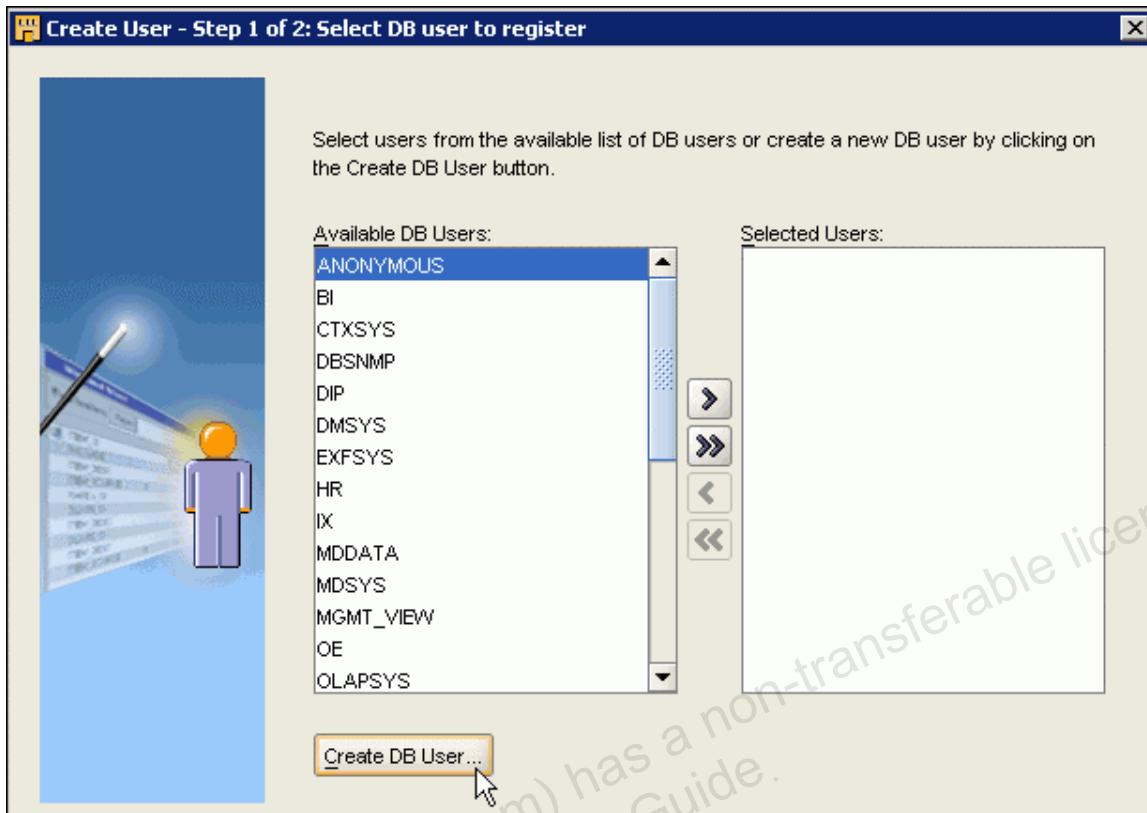
Password: **STAGING_USER**



Answer:

1. In the Global Explorer panel of the Design Center, expand the Security node. Right-click the Users node, and select New.
Note: The Warehouse Builder Error dialog box appears if you have not saved your changes before creating users. In the Design Center, select Save All from the Design menu. After the changes are saved, perform step 1.
2. The Create User Wizard is launched. Click Next on the Welcome page.
3. On the Select DB User to Register page, click Create DB User.

Solutions for Practice 5-1: Create an External Table (continued)



4. On the Create Database User page, specify the following details:

Specify the username and password with the SYSDBA privilege:

SYSDBA name: Accept the default (sys).

SYSDBA password: oracle

Provide information to create the new DB user:

Name: **STAGING_USER**

Password: **STAGING_USER**

Confirm Password: **STAGING_USER**

Table Space: Accept the default.

Default: USERS

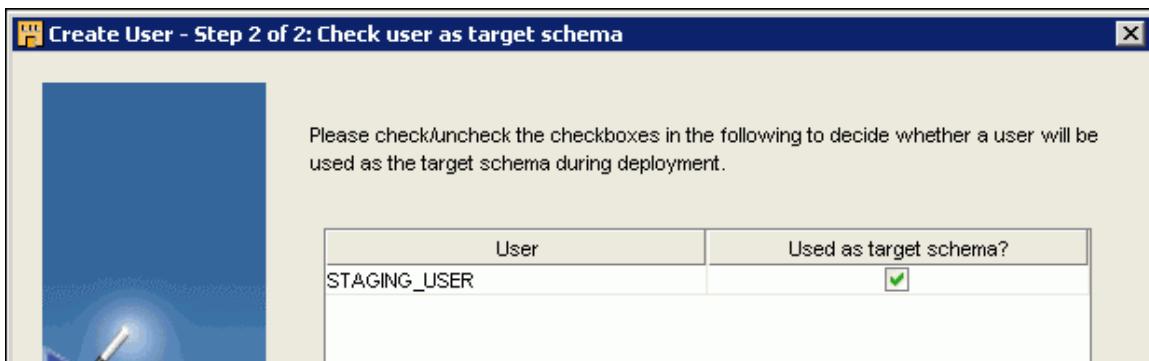
Temporary: TEMP

Click OK.

Solutions for Practice 5-1: Create an External Table (continued)



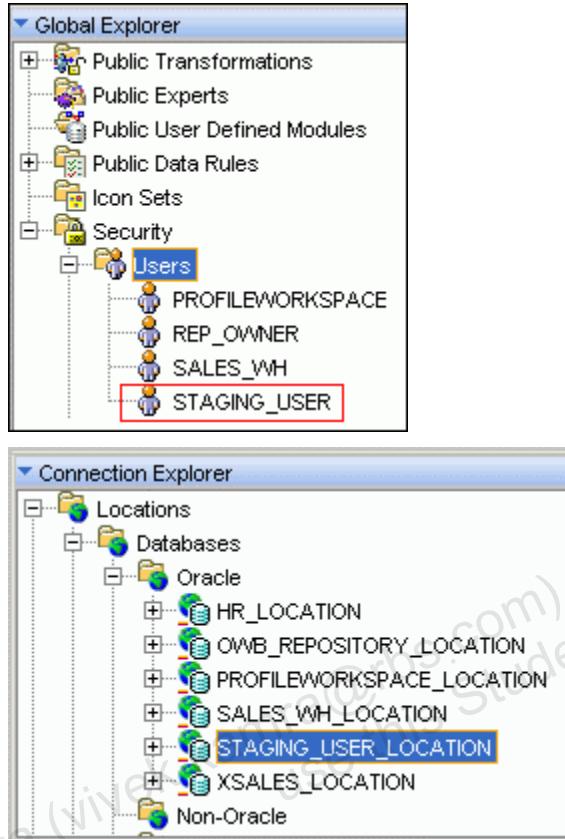
5. On the Select DB User to Register page, you now see STAGING_USER in the Selected Users list. Click Next.
6. On the Check User as Target Schema page, select the check box in the Used as Target Schema column for STAGING_USER. Click Next.



7. In the Target User Password dialog box, enter STAGING_USER as the password for the user that you identified as a target user, and click OK. Click Finish on the Summary page.

Solutions for Practice 5-1: Create an External Table (continued)

8. After you have completed all the wizard steps, a progress dialog box indicates that the user is being created. After the user has been successfully created, you see it in the Global Explorer in the Design Center. In the Connection Explorer panel, expand **Locations > Databases > Oracle**. STAGING_USER_LOCATION is created for you.



Create a Target Module

- 2) Create a target module named STAGING_AREA.

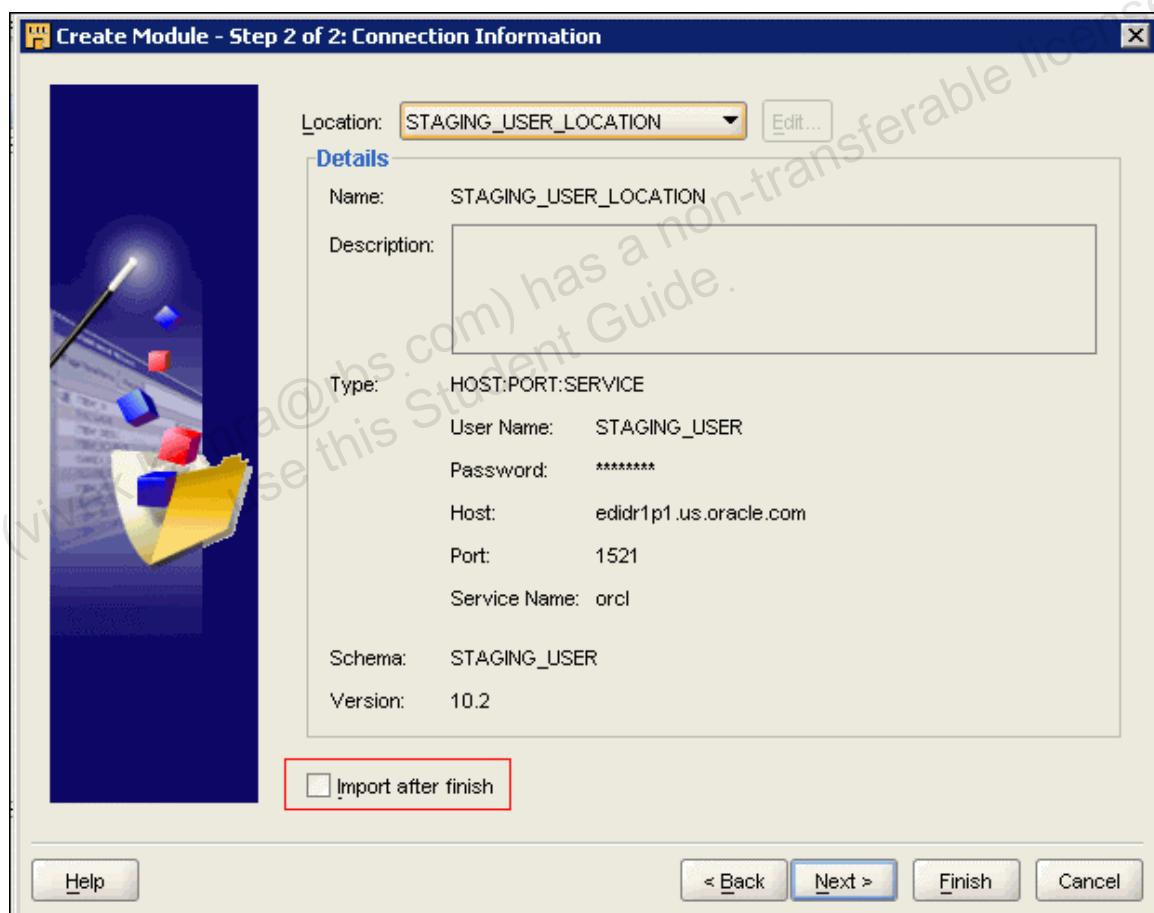
Answer:

1. In the Design Center, in the Project Explorer, expand **CLASS_PROJECT > Databases**. Right-click **Oracle**, and select **New**. The Create Module Wizard is launched. Click **Next** on the Welcome page.
2. On the Name and Description page, enter the following information:
 - a. Name: **STAGING_AREA**
 - b. Module status: Accept the default (Development)
 - c. Module type: Choose Warehouse Target
 - d. Description: This warehouse module is used to store the staging tablesClick **Next**.

Solutions for Practice 5-1: Create an External Table (continued)

3. On the Connection Information page, select **STAGING_USER_LOCATION** from the Location drop-down list. The details are automatically filled in. Review the details:
 - a. Type: HOST: PORT: SERVICE (default)
 - b. Username: staging_user
 - c. Host: localhost or <host name that you specified>
 - d. Port: 1521
 - e. Service: orcl or <your database service name>
 - f. Version: 10.2

Ensure that the “Import after finish” check box is not selected.



4. Click Next. The Summary page appears. Review the Summary page, and click Finish. The target module is now created.
- 3) Save your work.

Answer:

In the Design Center, click the Save All icon on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

Create an External Table

Solutions for Practice 5-1: Create an External Table (continued)

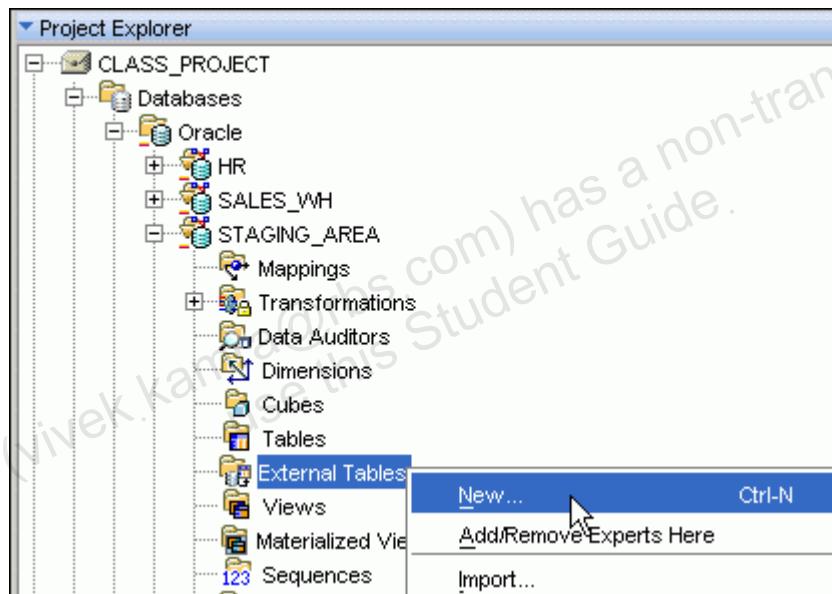
- 4) Under the STAGING_AREA target module, create the GEOG_EXT external table with the following details:

(Note: The external table represents data from nonrelational sources in a relational table format.)

- a) External table name: GEOG_EXT
- b) File name: FILE_SOURCE -> GEOGRAPHY.TXT
- c) Default Location: FILE_SOURCE_LOCATION

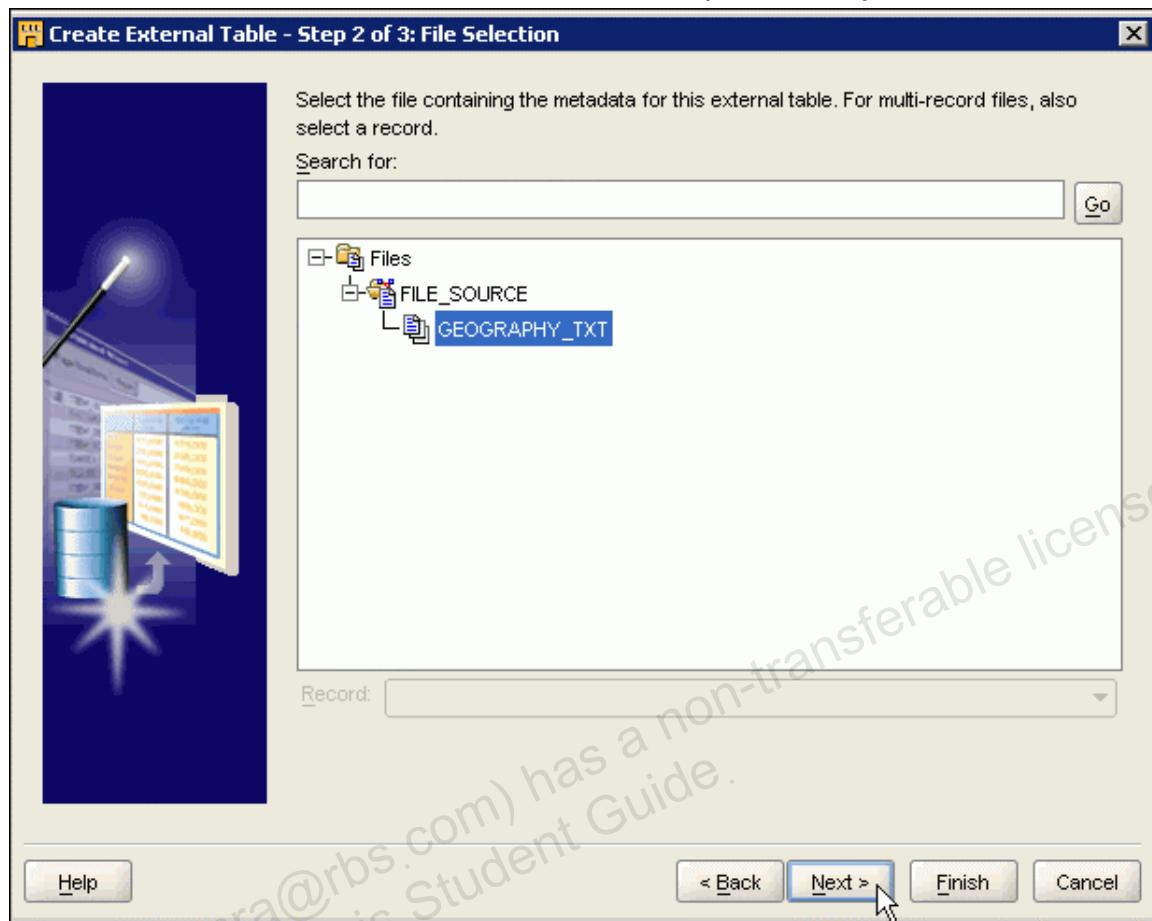
Answer:

- a. In the Project Explorer, expand CLASS_PROJECT. Expand Databases > Oracle > STAGING_AREA.
- b. Right-click External Tables, and select New. The Create External Table Wizard is launched. Click Next on the Welcome page.

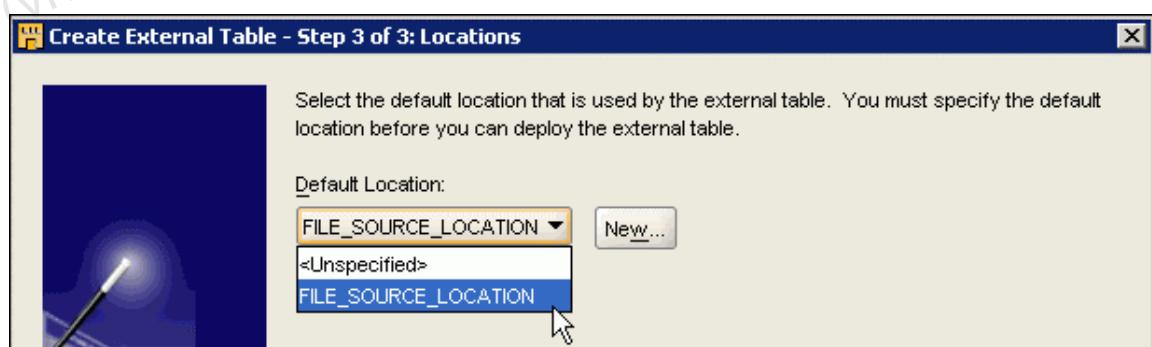


- c. On the Name and Description page, enter GEOG_EXT as the name for the external table. Click Next.
- d. On the File Selection page, expand FILES > FILE_SOURCE, and select GEOGRAPHY.TXT. Click Next.

Solutions for Practice 5-1: Create an External Table (continued)

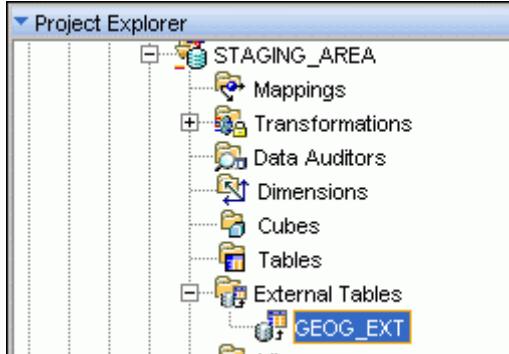


- e. On the Locations page, from the default Location drop-down list, select FILE_SOURCE_LOCATION. Click Next.



- f. Review the summary information, and click Finish.
- g. The external table is successfully created. In the Project Explorer, you see an entry for GEOG_EXT under the External Tables node, under STAGING_AREA.

Solutions for Practice 5-1: Create an External Table (continued)

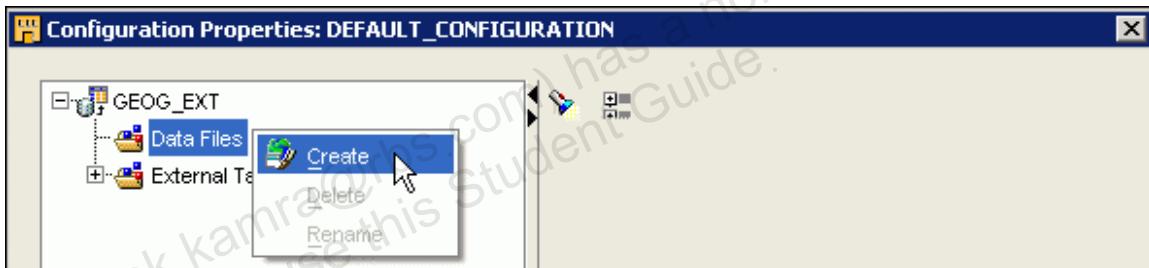


Configuration Setting for the External Table

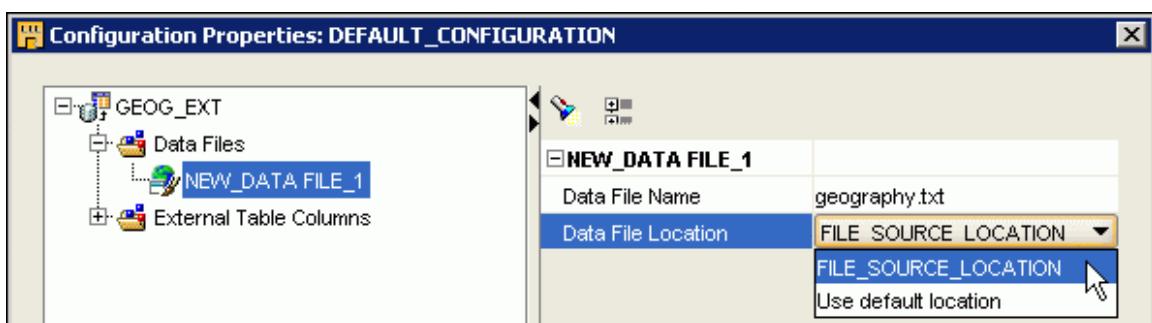
- 5) Configure the physical file system details for the external table you created.

Answer:

1. In the Project Explorer, expand STAGING_AREA > External Tables > GEOG_EXT. Right-click GEOG_EXT, and select Configure.
2. In the Configuration Properties window, expand GEOG_EXT (if not already expanded). Right-click Data Files, and select Create.



3. Accept the default name of NEW_DATAFILE_1. On the right, you see details to be entered for NEW_DATAFILE_1. Enter the following details:
 - a. Data File Name: geography.txt
 - b. Data File Location: FILE_SOURCE_LOCATIONClick OK.



You have completed configuring (assigning physical properties) the external table.

- 6) Save your work.

Answer:

Solutions for Practice 5-1: Create an External Table (continued)

In the Design Center, click Save All icon on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target

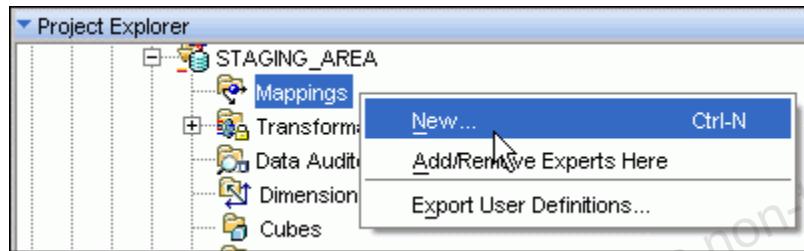
In this practice, you create your first mapping to map objects from the source to the target.

The mapping is designed to load data from the GEOG_EXT source external table to the STG_GEOG staging table.

- 1) Create a mapping named **LOAD_STG** to load data into the STG_GEOG staging table.

Answer:

1. In the Design Center, expand **CLASS_PROJECT > Databases > Oracle > STAGING_AREA**. Right-click the **Mappings** node, and select **New**.

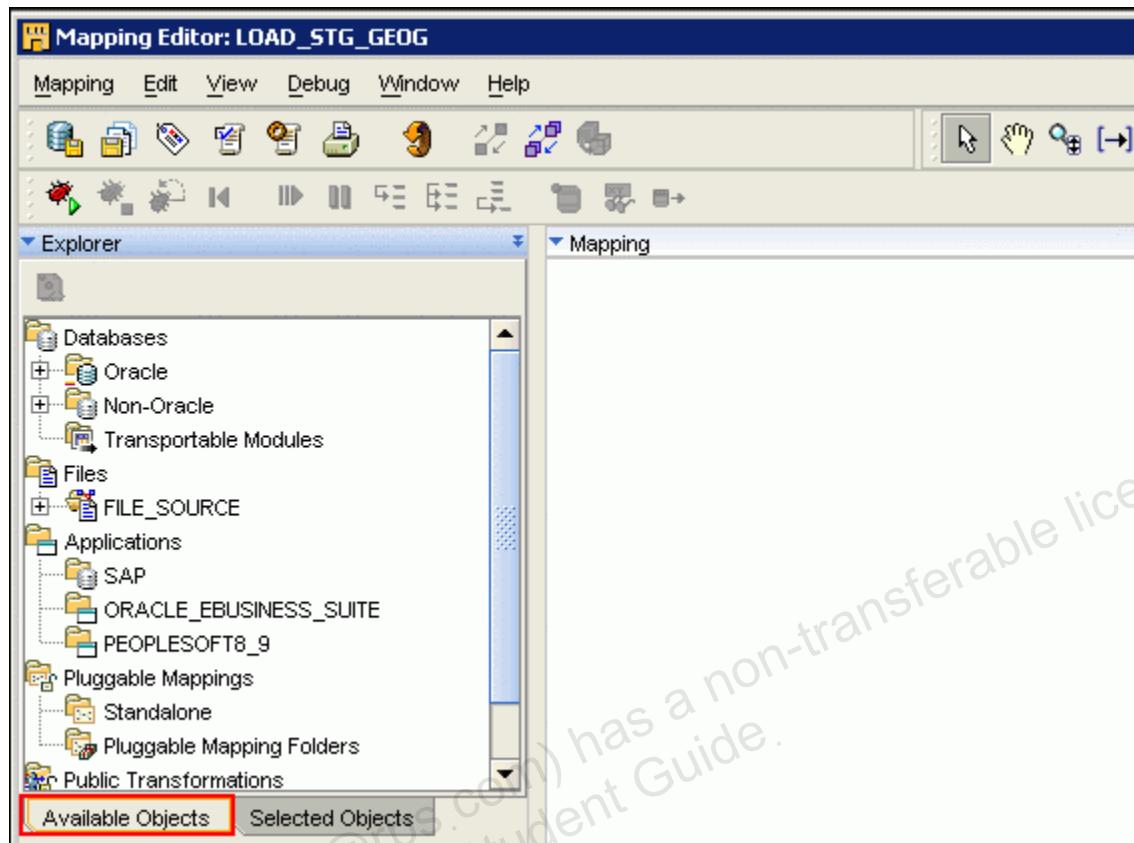


2. In the Create Mappings dialog box, enter **LOAD_STG_GEOG** as the mapping name. The Mapping Editor is launched.
- 2) Add the GEOG_EXT external table.

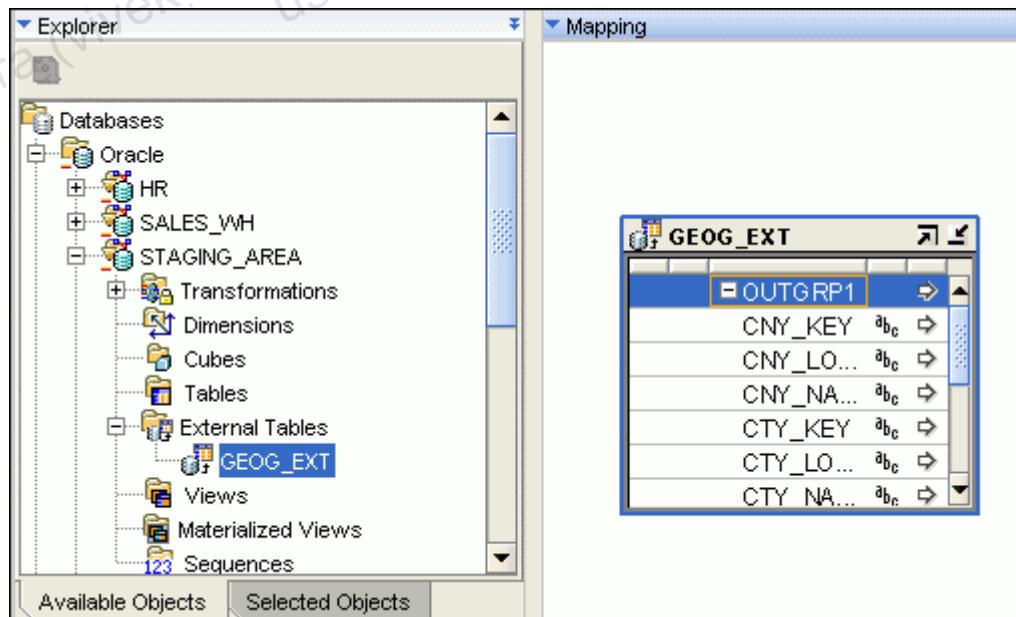
Answer:

1. In the Mapping Editor, in the Explorer panel, ensure that the Available Objects tabbed page is displayed.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



2. Expand Oracle > STAGING_AREA > External Tables. Drag GEOG_EXT to the canvas. On the canvas, you now see one object: the GEOG_EXT external table.

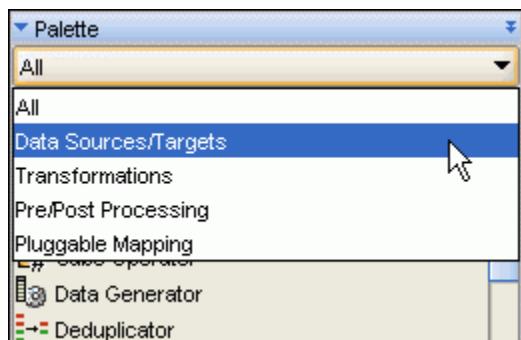


- 3) Create an unbound table and name it STG_GEOG.

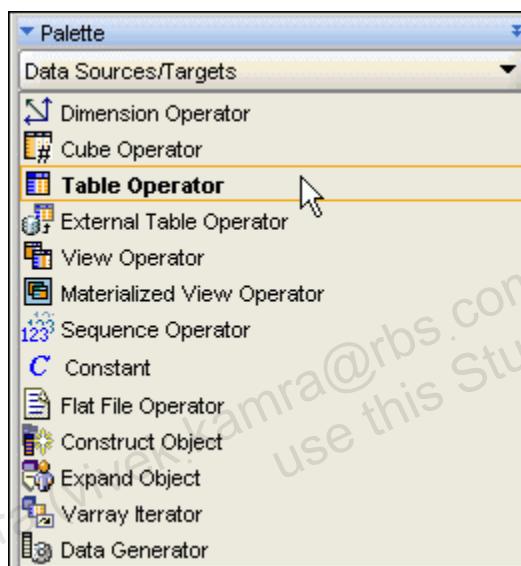
Answer:

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

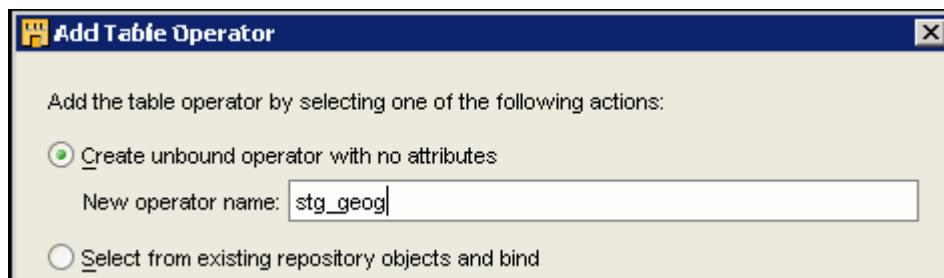
1. From the Palette, select Data Sources/Targets from the drop-down list.



2. From the list, drag Table Operator to the canvas. The Add Table Operator dialog box appears.



3. Select “Create unbound operator with no attributes.”
4. In the New operator name, enter STG_GEOG, and click **OK**.



5. On the canvas, you see two operators now: an external table operator GEOG_EXT and an unbound table operator STG_GEOG.

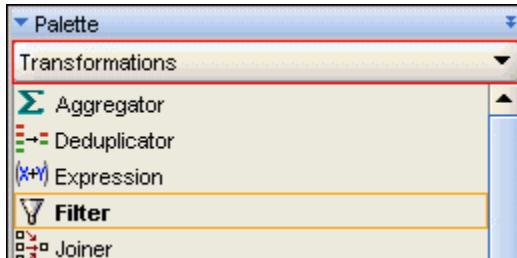
Add a Filter

- 4) Drag a filter operator named REGIONS to the canvas.

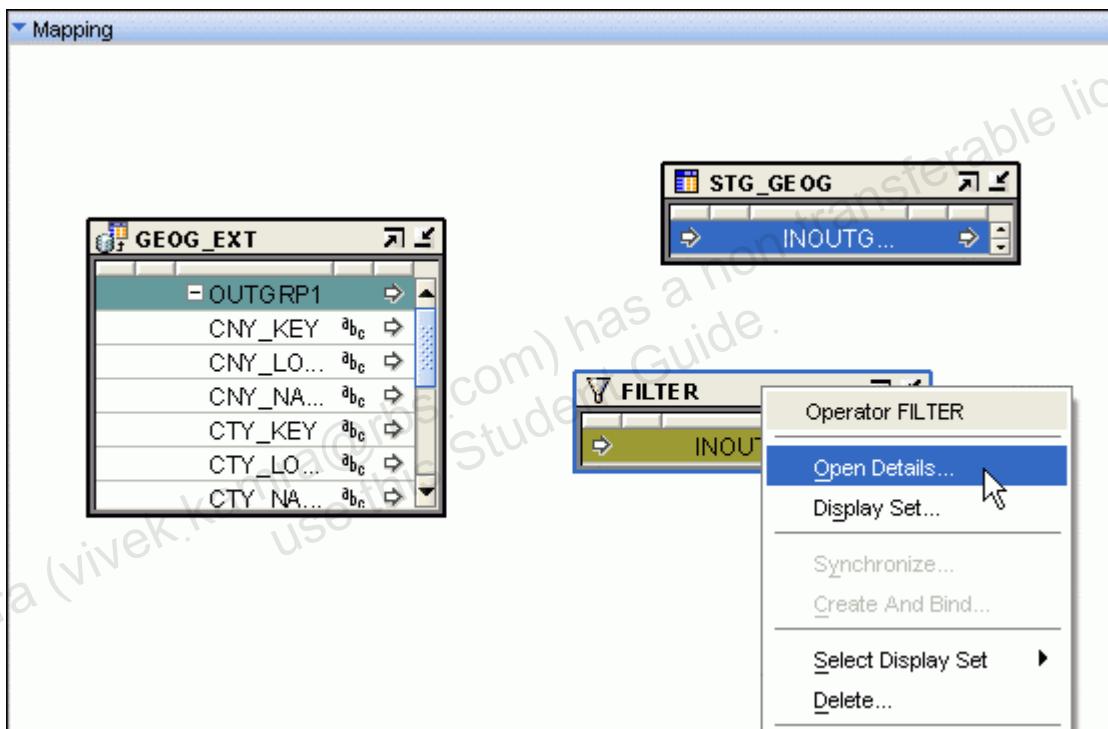
Answer:

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

- From the Palette, select Transformations from the drop-down list. Drag the Filter operator to the canvas.



- Right-click FILTER, and select Open Details. The Filter Editor appears.



- Change FILTER to REGIONS, and click OK.

On the canvas, you have three operators: an external table operator GEOG_EXT, a table operator STG_GEOG, and a filter operator REGIONS.

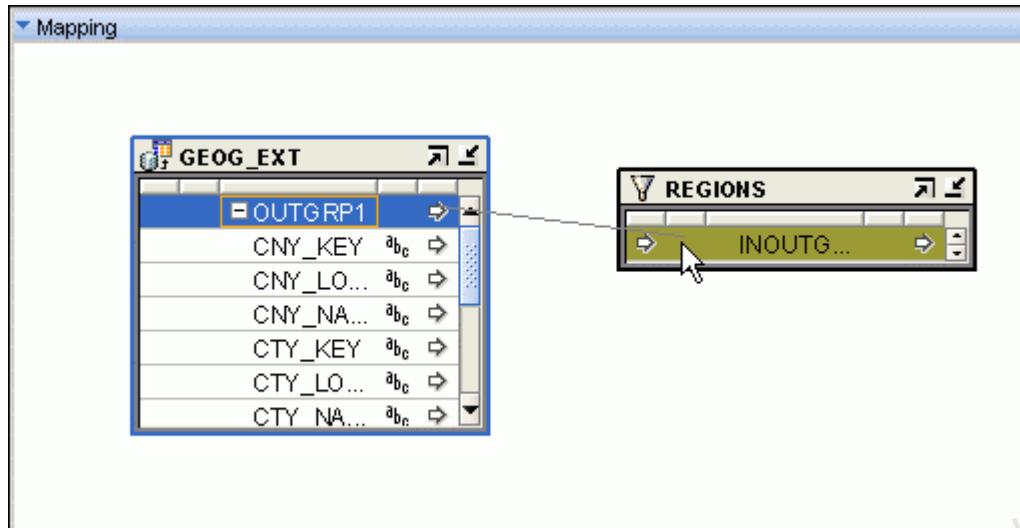
Connect Groups

- Connect OUTGRP1 of the GEOG_EXT external table operator to INOUTGRP1 of the REGIONS filter operator.

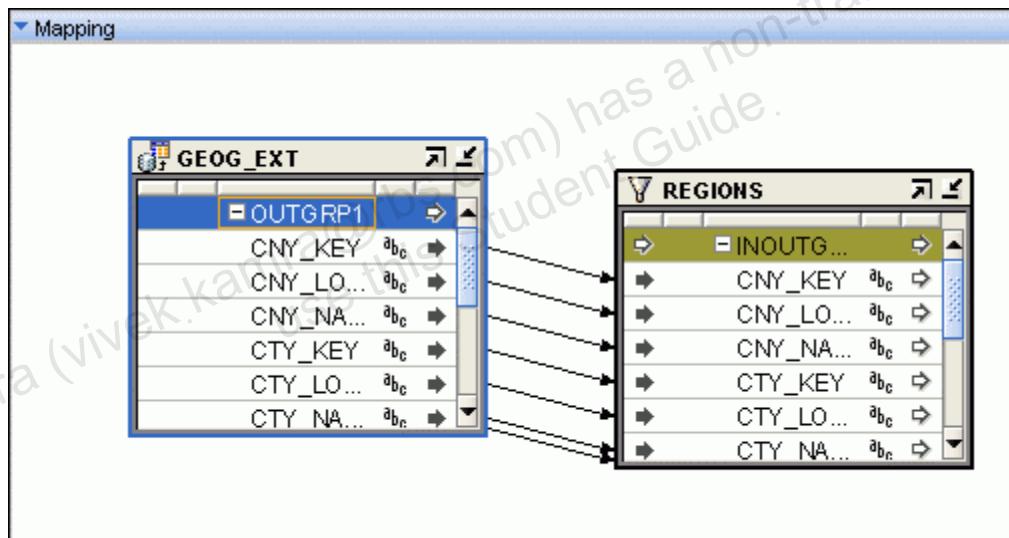
Answer:

- On the canvas, drag a line from OUTGRP1 of the GEOG_EXT external table operator to INOUTGRP1 of the REGIONS filter operator.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



2. All the attributes from the GEOG_EXT external table are mapped to REGIONS. Thus, in one quick step, you have mapped a group of attributes.

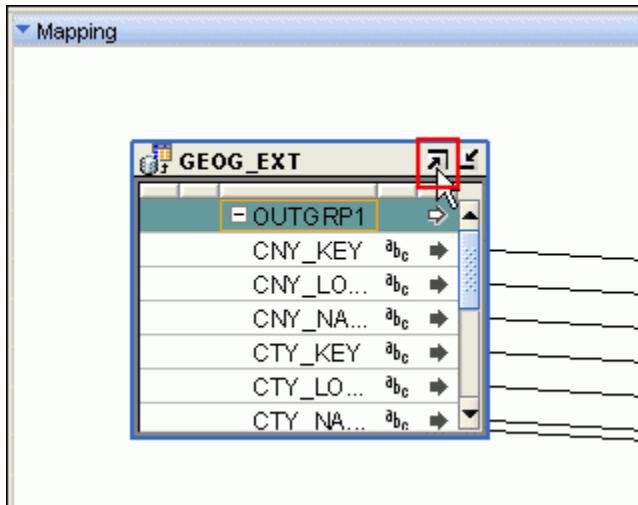


- 6) Maximize the GEOG_EXT external table operator to be able to see all the attributes.

Answer:

In the header area of the GEOG_EXT external table, click the maximize icon as shown in the following screenshot:

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



- 7) Save your work.

Answer:

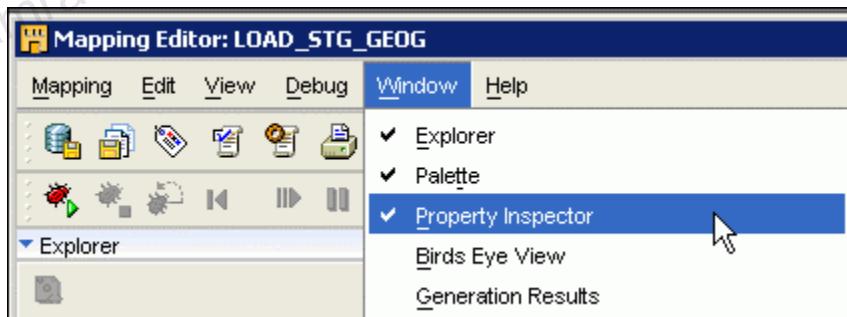
In the Mapping Editor, click the Save All icon on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

Define the Filter Expression

- 8) For the REGIONS filter, specify the following filter expression:

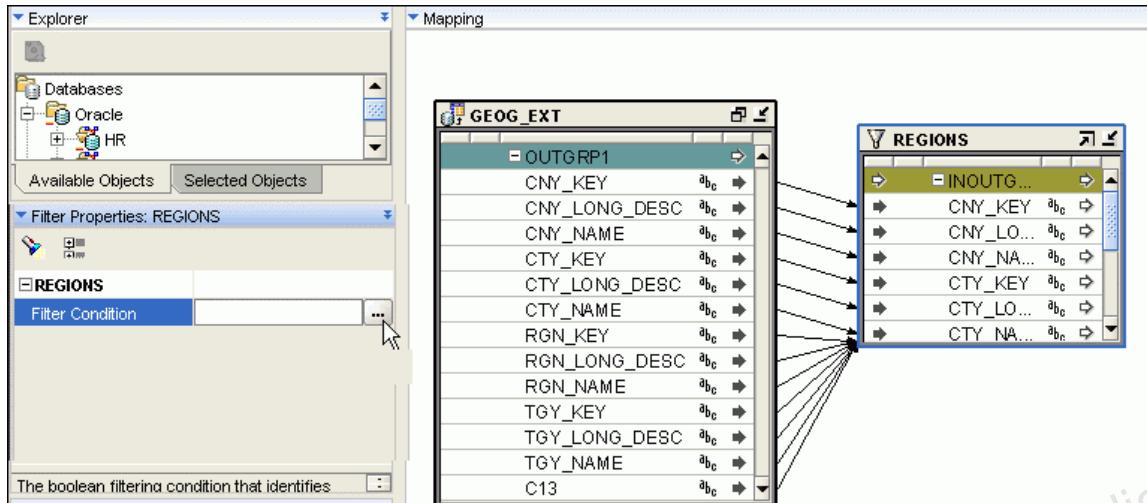
INOUTGRP1.RGN_LONG_DESC LIKE 'AMERICAS' (case sensitive)

- 1) In the Mapping Editor, ensure that Property Inspector is selected on the Window menu.



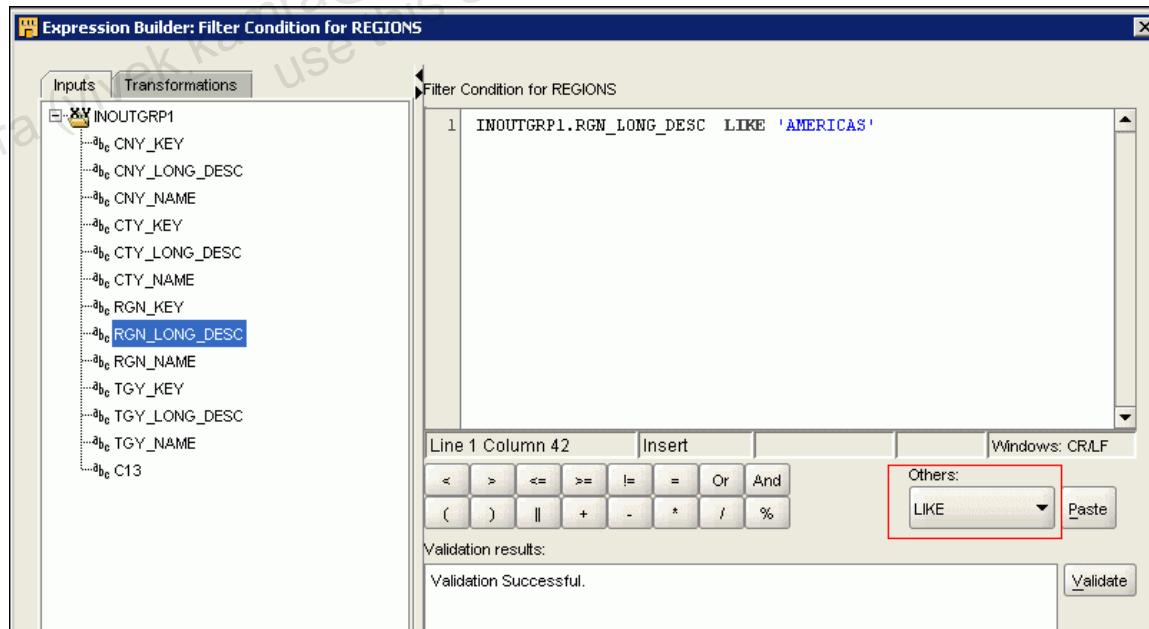
- 2) On the canvas, click the header area of the REGIONS operator.
- 3) On the left, in the Filter Properties panel, click the field next to the Filter Condition property. Click the three-dotted button (...) on the right. The Expression Builder is launched.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



- 4) In the Expression Builder, on the left, expand INOUTGRP1. Double-click RGN_LONG_DESC. This now appears on the right.
- 5) In the lower-right section of the Expression Builder, from the Others drop-down list, select LIKE, and click Paste. The Filter condition now reads:
INOUTGRP1.RGN_LONG_DESC LIKE
- 6) Now complete the filter condition by entering 'AMERICAS' (must be uppercase and within single quotation marks).

Click Validate. "Validation results" displays the validation results. Click OK.



You have specified the filter condition. Now connect REGIONS to the STG_GEOG staging target table.

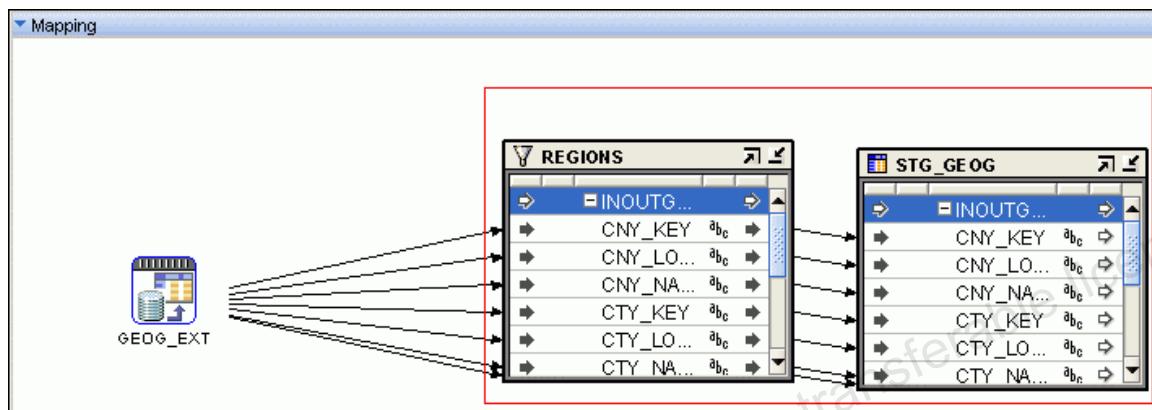
Connect REGIONS and STG_GEOG

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

- 9) Drag a connection line from INOUTGRP1 of REGIONS to STG_GEOG.

Answer:

On the canvas, drag a line from INOUTGRP1 of REGIONS to INOUTGRP1 of STG_GEOG. All the attributes from REGIONS are mapped to STG_GEOG. Thus, in one quick step, you have mapped a group of attributes.

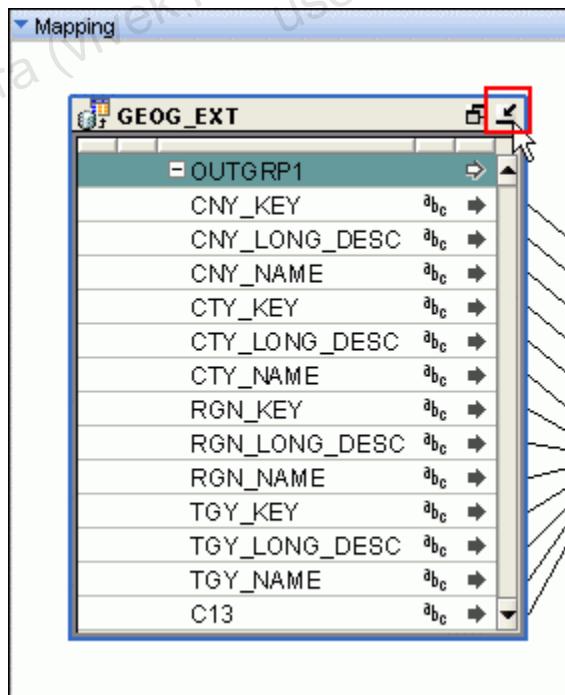


To be able to see GEOG_EXT as in this screenshot, minimize the GEOG_EXT external table operator to icon size.

- 10) Minimize the GEOG_EXT external table operator.

Answer:

1. On the canvas, select the GEOG_EXT external table operator, and click the minimize icon as shown in the screenshot



- 11) Create the STG_GEOG table in the repository.

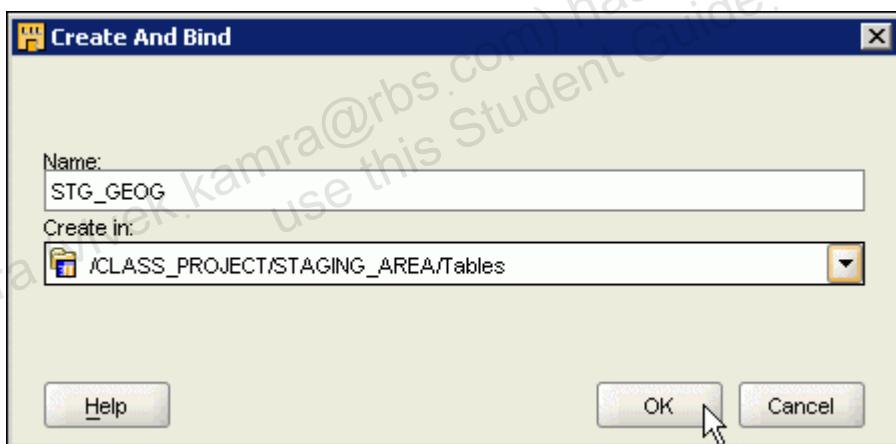
Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

Answer:

1. On the canvas, in the Mapping Editor, right-click STG_GEOG, and select Create and Bind. The Create and Bind dialog box appears.



2. The name is STG_GEOG (the same as that specified for the table operator).
3. Expand the “Create in” drop-down list, and select STAGING_AREA > TABLES. Click OK.



4. You have now successfully created a simple mapping that filters out those rows from the external table where the region is AMERICAS and load it into the STG_GEOG staging table.

Validate and Generate the Mapping in the Mapping Editor

- 12) Save your work.

Answer:

In the Mapping Editor, click Save All on the toolbar.

- 13) Validate the LOAD_STG_GEOG mapping.

Answer:

1. From the Mapping menu, select Validate.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



2. In the Generation Results panel, you see the results of the Validation.

Generation Results			
Message			
	Code	Message	Validation Details
	<input checked="" type="checkbox"/> Success	VLD-0001: Validation completed successfully.	Validation completed successfully.

- 14) Generate the LOAD_STG_GEOG mapping.

Note: When you generate a mapping, OWB implements the SQL code to perform DML and DDL commands that are necessary to move data from the sources to the mapped target.

Answer:

1. In the Mapping Editor, select Generate from the Mapping menu.
2. The Generation Results window displays the generated code.

Generation Results	
Generation style:	Full
Operating mode:	[PL/SQL] SET_...
Script	Message
53	"CTY_KEY",
54	"CTY_LONG_DESC",
55	"CTY_NAME",
56	"RGN_KEY",
57	"RGN_LONG_DESC",
58	"RGN_NAME",
59	"TGY_KEY",
60	"TGY_LONG_DESC",
61	"TGY_NAME",
62	"C13")

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

15) Close the Mapping Editor

Answer:

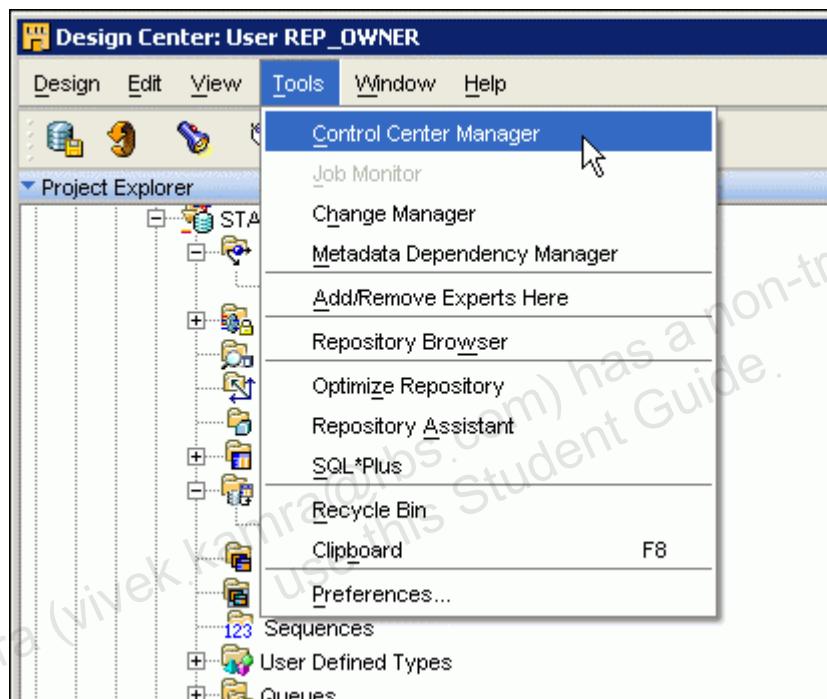
From the Mapping menu, select Close.

To see the results of your work, deploy the mapping and then execute the mapping.

16) Launch the Control Center Manager.

Answer:

1. In the Design Center, from the Tools menu, select Control Center Manager.

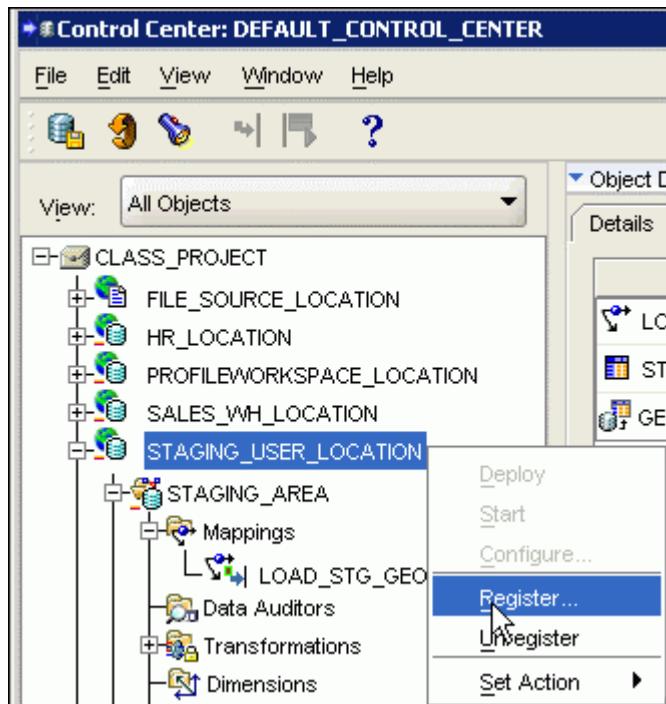


17) Register the STAGING_USER_LOCATION target location.

Answer:

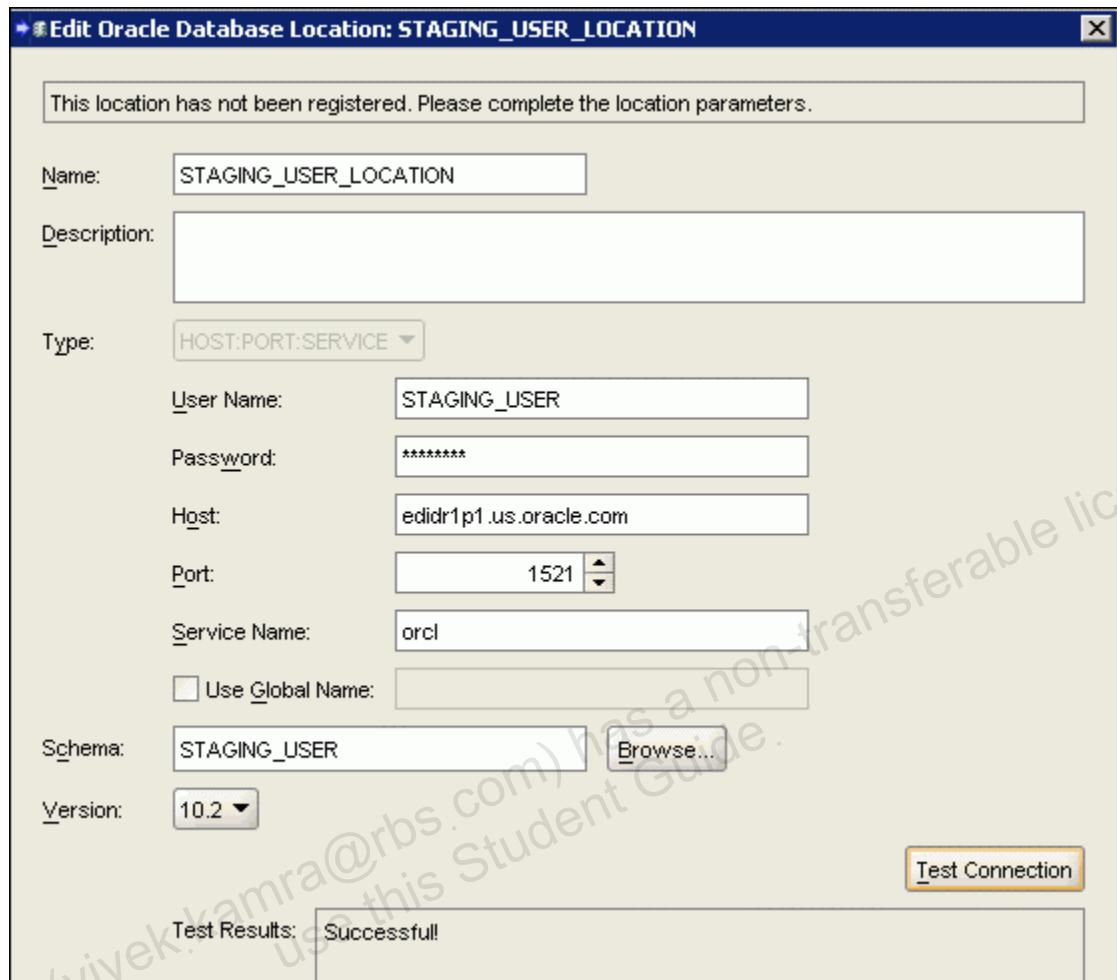
1. In the Control Center on the left in the navigation window, expand CLASS_PROJECT. Right-click STAGING_USER_LOCATION, and select Register.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



2. The Edit Oracle Database Location dialog box appears. Click Test Connection. If the connection is successful, click OK. The value in the Host field can be localhost or <name of your computer>. Do not try to enter the value as you see in the screenshot.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



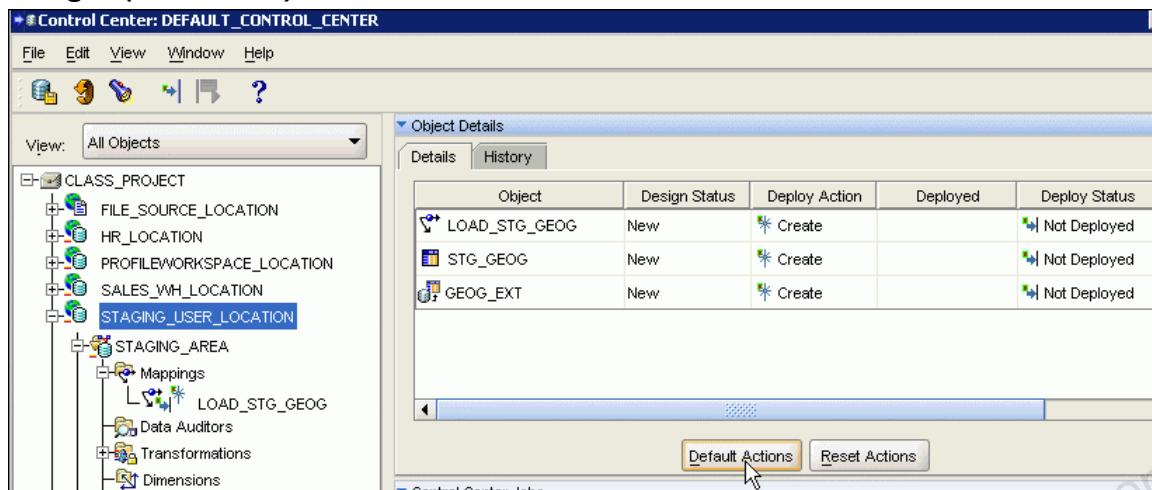
- 18) Deploy the LOAD_STG_GEOG mapping and the dependent objects.

Answer:

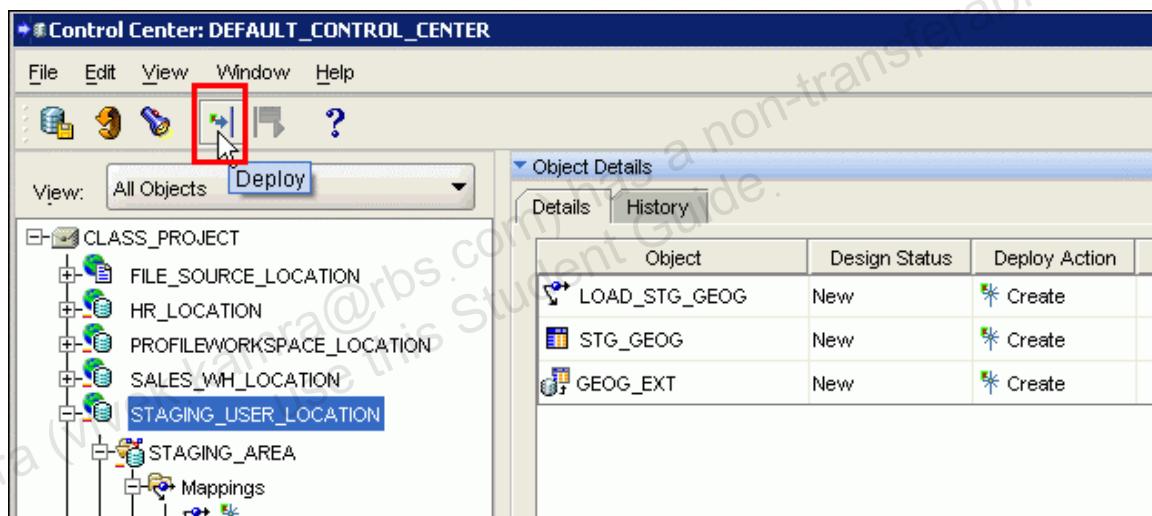
1. In the Control Center, click Default Action.

Note: The value in the Deploy Action column changes from None to Create.

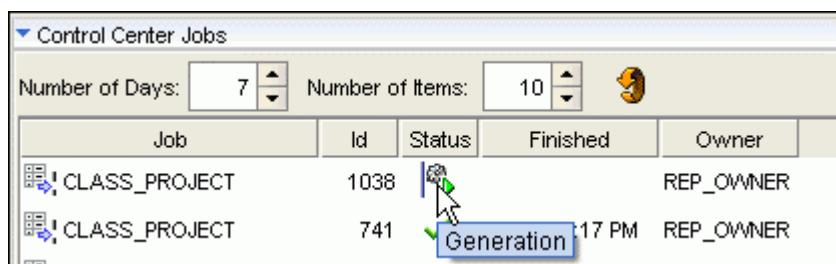
Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



2. Click Deploy on the toolbar.



3. In the Control Center Jobs panel, monitor the progress.



4. The deployment completes successfully.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

The screenshot shows the Oracle Control Center Manager interface. The top section, 'Object Details', displays a table of objects with columns: Object, Design Status, Deploy Action, Deployed, and Deploy Status. Three rows are listed: LOAD_STG_GEOG (Success), STG_GEOG (Success), and GEOG_EXT (Success). The Deploy Status column is highlighted with a red border. Below this is a toolbar with 'Default Actions' and 'Reset Actions'. The bottom section, 'Control Center Jobs', shows a table with columns: Job, Id, Status, Finished, and Owner. One job is listed: CLASS_PROJECT (Id: 1038, Status: ✓, Finished: 5/4/06 8:37 PM, Owner: REP_OWNER). This row is also highlighted with a red border.

Object	Design Status	Deploy Action	Deployed	Deploy Status
LOAD_STG_GEOG	Unchanged	None	5/4/06 8:37 PM	Success
STG_GEOG	Unchanged	None	5/4/06 8:37 PM	Success
GEOG_EXT	Unchanged	None	5/4/06 8:37 PM	Success

Job	Id	Status	Finished	Owner
CLASS_PROJECT	1038	✓	5/4/06 8:37 PM	REP_OWNER

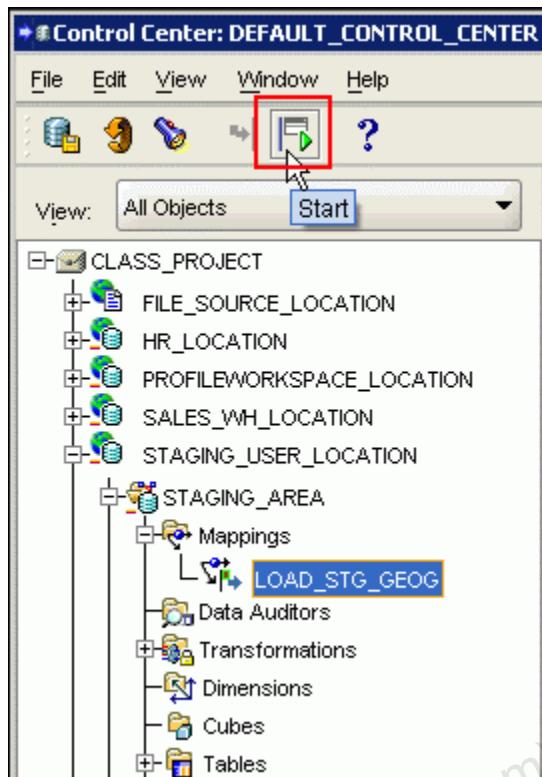
You now execute the mapping that will load the data into the STG_GEOG table.

- 19) Execute the LOAD_STG_GEOG mapping.

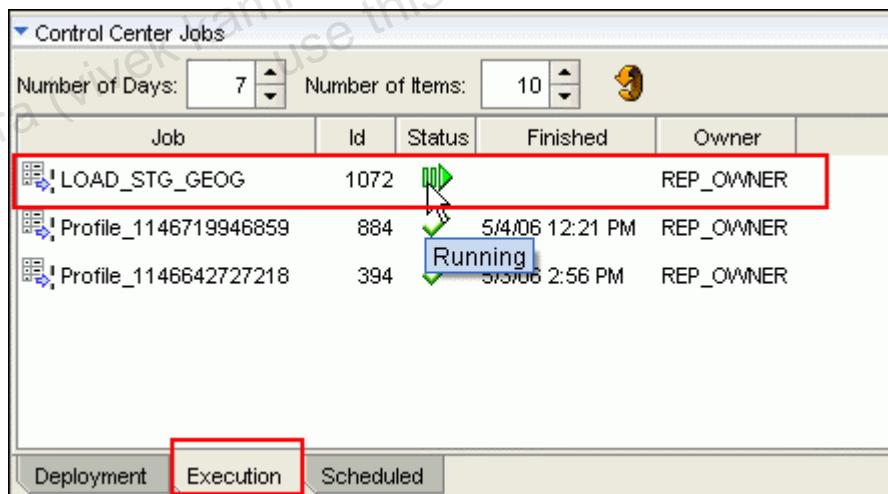
Answer:

1. In the Control Center Manager, select the LOAD_STG_GEOG mapping, and click the Start icon on the toolbar.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)



2. In the Control Center Jobs panel, monitor the progress of the execution.



When the execution completes successfully, you are ready to view the data.

- 20) Close the Control Center Manager.
21) View the data in the STG_GEOG table.

Answer:

In the Design Center, right-click STG_GEOG, and select Data.

Solutions for Practice 5-2: Create a Simple Mapping from Source Objects to the Target (continued)

The screenshot shows the Oracle Warehouse Builder interface. The Project Explorer on the left lists various database objects under the CLASS_PROJECT. A context menu is open over the STG_GEOG table, with the 'Data...' option highlighted. Below the Project Explorer is the Relational Data Viewer window, which displays a grid of data for the STG_GEOG table. The last two columns of the viewer are highlighted with a red border.

CNY_KEY	CNY_LONG_...	CNY_NAME	CTY_KEY	CTY_LONG_...	CTY_NAME	RGN_KEY	RGN_LONG_DESC	RGN_NAME	TGY_KEY	TGY_LONG_...
1	COLOMBIA	Colombia	BOGOTA	Bogota	Colombia	Bogota	AMERICAS	Areas in the ... Americas		WORLD
2	MEXICO	Mexico	MEXICOCITY	Mexico City	Mexico	Mexico City	AMERICAS	Areas in the ... Americas		WORLD
3	USA	United State...	USA	NEWYORK	New York	New York	NYC	AMERICAS	Areas in the ... Americas	WORLD
4	ARGENTINA	Argentina	BUENOSAIRES	Buenos Aires	Argentina	Buenos Aires	AMERICAS	Areas in the ... Americas		WORLD
5	USA	United State...	USA	BOSTON	Boston	Massachus...	Boston	AMERICAS	Areas in the ... Americas	WORLD
6	USA	United State...	USA	LOSANGELES	Los Angeles	California	LA	AMERICAS	Areas in the ... Americas	WORLD
7	USA	United State...	USA	DENVER	Denver	Colorado	Denver	AMERICAS	Areas in the ... Americas	WORLD
8	USA	United State...	USA	DALLAS	Dallas	Texas	Dallas	AMERICAS	Areas in the ... Americas	WORLD
9	BRAZIL	Brazil	SAOPAULO	Sao Paulo	Brazil	Sao Paulo	AMERICAS	Areas in the ... Americas		WORLD
10	USA	United State...	USA	CHICAGO	Chicago	Illinois	Chicago	AMERICAS	Areas in the ... Americas	WORLD
11	USA	United State...	USA	SEATTLE	Seattle	Washington	Seattle	AMERICAS	Areas in the ... Americas	WORLD

22) Close the Relational Data Viewer.

23) Save your work.

Practice Solutions for Lesson 6

In this practice, you learn how to derive data rules and run correction mappings.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings

Now that you are familiar with creating a target module, location, and target user, you run this script that will create the target user DP_TGT, a target module DP_TGT, and a location DP_TGT_LOCATION for you. Also, the DP_TGT target user is granted privileges to access tables from the HR schema.

- 1) Run the `create_dp_tgt.tcl` file in the OMB*Plus window in the Design Center.

Answer:

1. From the Window menu in the Design Center, select OMB*Plus.



2. In the OMB*Plus window, enter the following commands at the OMB prompt:
OMB+> cd e:/labs/tcl
OMB+> source create_dp_tgt.tcl

The screenshot shows the OMB*Plus window with the command-line interface. The history pane shows the following commands:
OMB+> cd e:/labs/tcl
OMB+> source create_dp_tgt.tcl
User DP_TGT registered.
Context changed.
Oracle module DP_TGT created.
DP_TGT Target module created
Commit complete.
OMB+> |

- 2) Open the DP_HR data profile.

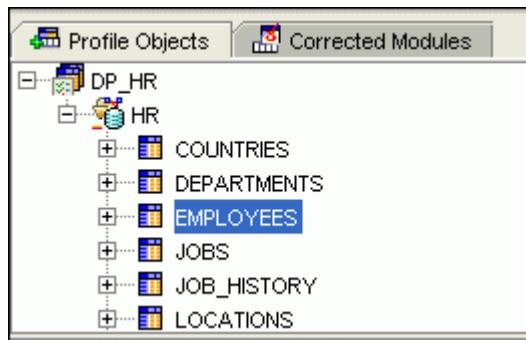
Answer:

1. In the Project Explorer, expand CLASS_PROJECT > Data Profiles, and double-click DP_HR. The Data Profile Editor opens.
- 3) Display the Domain values for JOB_ID in the EMPLOYEES table.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

Answer:

1. In the Data Profile Editor, select EMPLOYEES from the navigation tree on the left.



2. In the Profile Results Canvas window, click the Domain tab.
3. In the Columns column, scroll down to JOB_ID, and click the hyperlink in the Found Domain column.
4. Notice the Data Drill Panel showing all the values (Compliant and Noncompliant).

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

Profile Results Canvas

Here are the domain analysis results for EMPLOYEES, which has 12 columns and 109 rows.

	Columns	Found Domain	% Compliant
	HIRE_DATE	.	0%
	JOB_ID	PU_CLERK SH_CLERK ST_CLERK AD_VP SA_REP	89%
	LAST_NAME	.	0%
	MANAGER_ID	120 100 101 145 146 123 149 121 108 122...	96.3%
	PHONE_NUJMR		0%

Derive Data

Data Drill Panel

Here are drill results on EMPLOYEES column JOB_ID related to Domains.

Distinct values:	All	Rows for the selected distinct value:																																										
<table border="1"> <thead> <tr> <th>JOB_ID</th> <th># Rows</th> <th>% of 109</th> </tr> </thead> <tbody> <tr><td>1 AC_ACCOUNT</td><td>1</td><td>.9%</td></tr> <tr><td>2 AC_MGR</td><td>1</td><td>.9%</td></tr> <tr><td>3 SA_REPP</td><td>1</td><td>.9%</td></tr> <tr><td>4 PU_CLIRK</td><td>1</td><td>.9%</td></tr> <tr><td>5 PU_MAN</td><td>1</td><td>.9%</td></tr> <tr><td>6 AD_ASST</td><td>1</td><td>.9%</td></tr> <tr><td>7 MK_MAN</td><td>1</td><td>.9%</td></tr> <tr><td>8 PR_REP</td><td>1</td><td>.9%</td></tr> <tr><td>9 FI_MGR</td><td>1</td><td>.9%</td></tr> <tr><td>10 MK_REP</td><td>1</td><td>.9%</td></tr> <tr><td>11 AD_PRES</td><td>1</td><td>.9%</td></tr> <tr><td>12 HR_REP</td><td>1</td><td>.9%</td></tr> <tr><td>13 AD_VP</td><td>2</td><td>1.8%</td></tr> </tbody> </table>	JOB_ID	# Rows	% of 109	1 AC_ACCOUNT	1	.9%	2 AC_MGR	1	.9%	3 SA_REPP	1	.9%	4 PU_CLIRK	1	.9%	5 PU_MAN	1	.9%	6 AD_ASST	1	.9%	7 MK_MAN	1	.9%	8 PR_REP	1	.9%	9 FI_MGR	1	.9%	10 MK_REP	1	.9%	11 AD_PRES	1	.9%	12 HR_REP	1	.9%	13 AD_VP	2	1.8%	All	Rows for the selected distinct value:
JOB_ID	# Rows	% of 109																																										
1 AC_ACCOUNT	1	.9%																																										
2 AC_MGR	1	.9%																																										
3 SA_REPP	1	.9%																																										
4 PU_CLIRK	1	.9%																																										
5 PU_MAN	1	.9%																																										
6 AD_ASST	1	.9%																																										
7 MK_MAN	1	.9%																																										
8 PR_REP	1	.9%																																										
9 FI_MGR	1	.9%																																										
10 MK_REP	1	.9%																																										
11 AD_PRES	1	.9%																																										
12 HR_REP	1	.9%																																										
13 AD_VP	2	1.8%																																										

Displaying 17 Rows out of 17

No Query Processed

- 4) Derive a data rule for this detected domain.

Answer:

1. If not already done, scroll down to the JOB_ID row. Select the values in the Found Domain column.
2. The symbol in the Found Domain column indicates that you can derive a data rule from the finding.
3. At the bottom right of the Profile Results Canvas, click Derive Rule.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

The screenshot shows the Oracle Warehouse Builder Profile Results Canvas. At the top, there are tabs for Domain, Unique Key, Functional Dependency, Referential, Data Rule, Data Profile, Profile Object, Aggregation, and Data Type. The Data Rule tab is selected. Below the tabs, a message states: "Here are the domain analysis results for EMPLOYEES, which has 12 columns and 109 rows." A table follows, showing domain analysis for various columns:

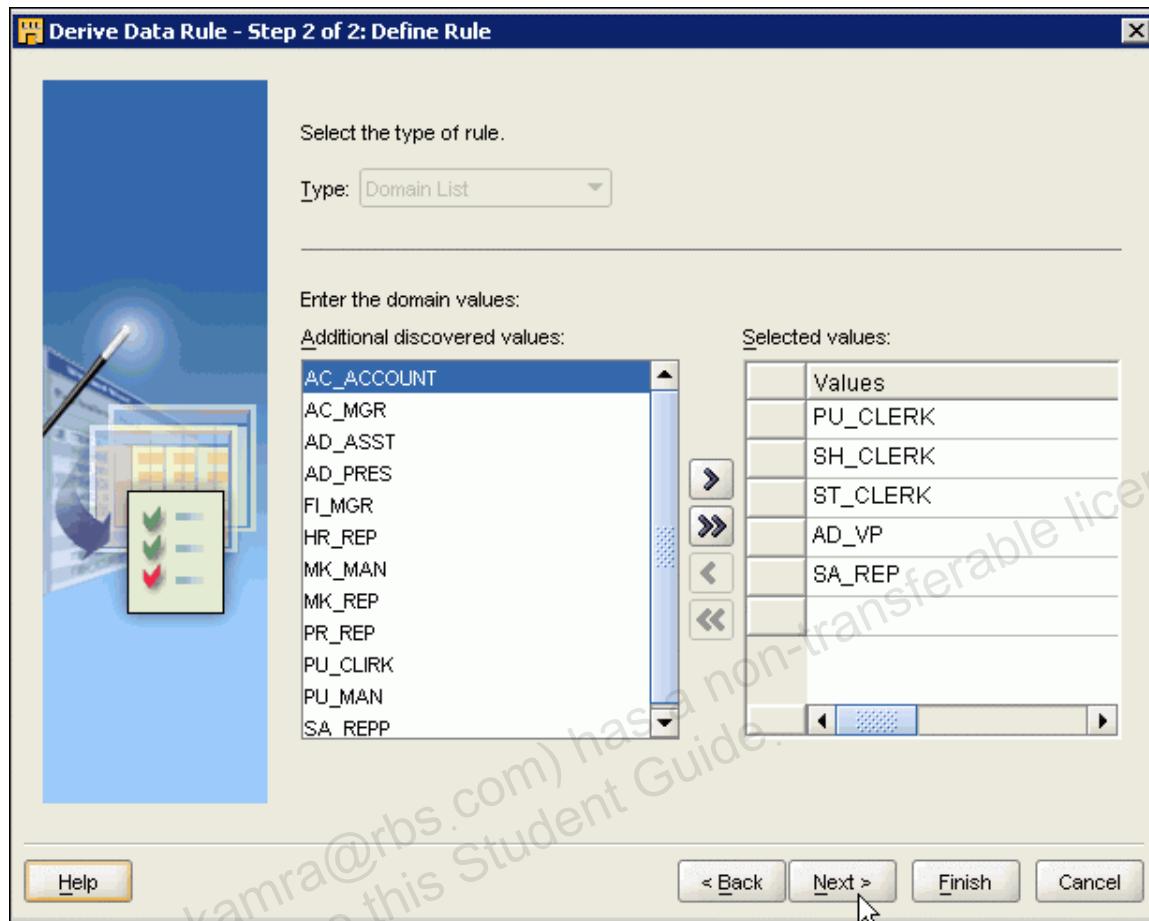
	Columns	Found Domain	% Compliant	Six Sigma Score
HIRE_DATE	.		0%	-6.2
JOB_ID	PU_CLERK SH_CLERK ST_CLERK AD_VP SA_REP		89%	2.73
LAST_NAME	.		0%	-6.2
MANAGER_ID	120 100 101 145 146 123 149 121 108 122...		96.3%	3.29
PHONE_NUMR			0%	-6.2

At the bottom right of the table area is a button labeled "Derive Data Rule". Below the table are two tabs: "Tabular" and "Graphical".

Note: You should have selected the values in the Found Domain column (as in step 1). If not, the Derive Rule button remains disabled.

4. The Derive Data Rule Wizard is launched. Click Next on the Welcome page.
5. On the Name and Description page, accept the default name for the Data Rule, and click Next.
6. On the Define Rule page, verify that the domain list is as shown in the screenshot, and click Next.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)



Note: You can, shuttle nonqualifying values across if you feel that they should be part of the domain. You can also enter new values in the list on the right if you know of other domain values not in your current data set.

- 7) Click Next. Review the Summary details, and click Finish.
- 8) Note that the Data Rule icon in the Found Domain column for JOB_ID changes from blue to green to indicate that a rule has been created for this column.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

Profile Results Canvas

Columns	Found Domain	% Compliant
EMPLOYEE_...	.	0%
FIRST_NAME	.	0%
HIRE_DATE	.	0%
JOB_ID	PU_CLERK SH_CLERK ST_CLERK AD_VP SA_REP	89%
LAST_NAME	.	0%
MANAGER_ID	120 100 101 145 146 123 140 121 108 122	96.3%

- 9) In the Data Rule Panel, you see a newly created data rule for JOB_ID.

Note: In the Data Rule Panel, scroll to the right to the Type column. For JOB_ID, Type is Domain List, whereas the others are custom rules, a rule type that is derived by profiling from the constraints on the Employee table.

Data Rule Panel

Applied Rules:				
	Name	Rule	Type	
<input checked="" type="checkbox"/>	EMP_SALARY_MIN	DERIVED_DATA_RULE...	Custom	
<input checked="" type="checkbox"/>	EMP_EMAIL_NN	DERIVED_DATA_RULE...	Custom	
<input checked="" type="checkbox"/>	EMP_JOB_NN	DERIVED_DATA_RULE...	Custom	
<input checked="" type="checkbox"/>	EMP_HIRE_DATE_NN	DERIVED_DATA_RULE...	Custom	
<input checked="" type="checkbox"/>	EMP_LAST_NAME_NN	DERIVED_DATA_RULE...	Custom	
<input checked="" type="checkbox"/>	JOB_ID	DERIVED_DATA_RULE...	Domain List	

After creating the rule, how will you use it? By creating a correction mapping.

Warehouse Builder not only shows you data and problems, and derivation of data rules and correction mappings, but it also enables you to act on this information. You will create correction mappings for cleansing Employees based on the data rules applied.

- 5) Generate a correction mapping for the JOB_ID rule that you created.

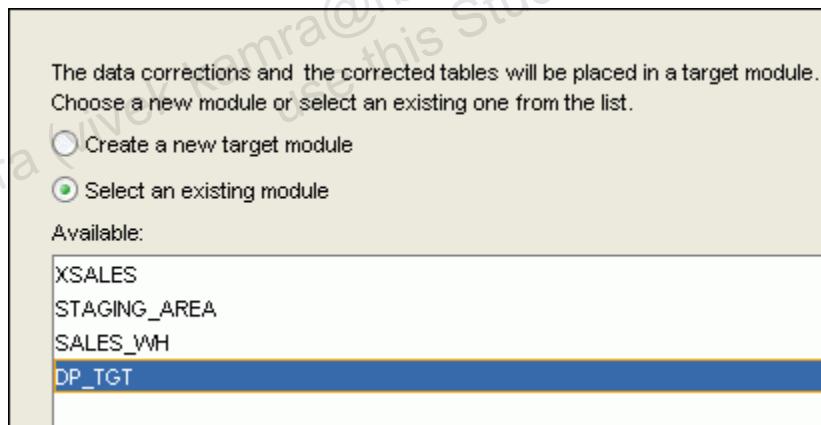
Answer:

1. In the Data Rule Panel, navigate to the JOB_ID row. Ensure that only the JOB_ID rule is selected and all others are deselected. This is important because you want to create a correction mapping only for the JOB_ID rule and not for the other rules.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

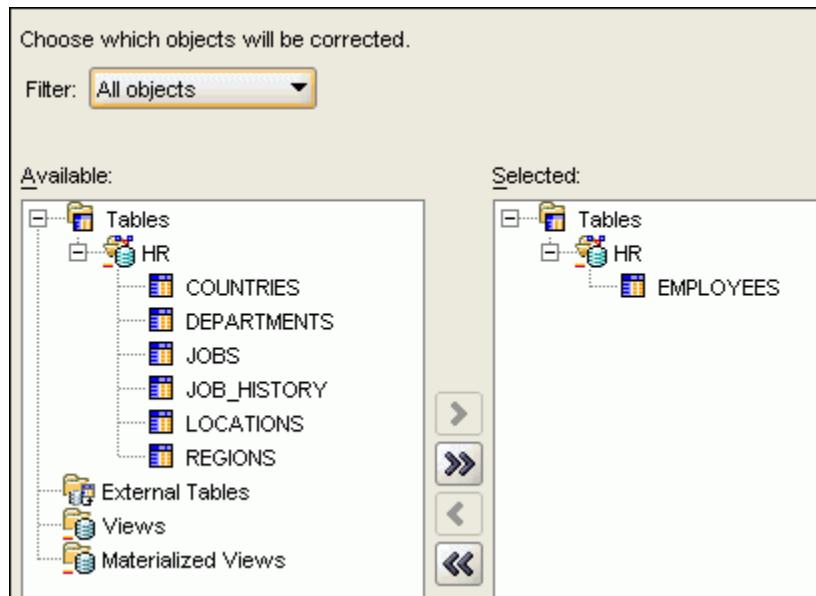
Data Rule Panel				
Applied Rules:				
		Name	Rule	Type
	<input type="checkbox"/>	EMP_SALARY_MIN	DERIVED_DATA_RULE...	Custom
	<input type="checkbox"/>	EMP_EMAIL_NN	DERIVED_DATA_RULE...	Custom
	<input type="checkbox"/>	EMP_JOB_NN	DERIVED_DATA_RULE...	Custom
	<input type="checkbox"/>	EMP_HIRE_DATE_NN	DERIVED_DATA_RULE...	Custom
	<input type="checkbox"/>	EMP_LAST_NAME_NN	DERIVED_DATA_RULE...	Custom
	<input checked="" type="checkbox"/>	JOB_ID	DERIVED_DATA_RULE...	Domain List

2. In the Data Profile Editor, select Create Correction from the Profile menu. The Create Correction Wizard is launched. Click Next on the Welcome page.
3. On the Select Target Module page, choose “Select an existing module” (if not already selected), and from the Available list, select DP_TGT. The DP_TGT schema was created for you as part of the setup. Click Next.



4. On the Select Objects page, select Tables from the Filter drop-down list. From the Available list, select and shuttle the EMPLOYEES tables to the Selected list if not already selected. Also, check to see that only the EMPLOYEES table is selected in the Selected list, and click Next.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)



- On the Select Data Rules and Data Types page, ensure that on the left, EMPLOYEES is selected, and on the right, the Data Rules tabbed page is displayed.

Create Correction - Step 3 of 7: Select Data Rules and Data Types

	Name	Rule	Type
<input checked="" type="checkbox"/>	JOB_ID	DERIVED_DATA_RULES.JOB_ID	Domain List

Bindings:

Parameter	Binding
VALUE	abc

- Navigate to the row where JOB_ID is displayed under the name column, and choose the JOB_ID rule by selecting check box to the left of JOB_ID. Note the Bindings section in the lower section. Click Next.
- On the Verify and Accept corrected tables page, examine the definition of the corrected table, and click Next.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

8. On the “Choose data correction actions” page, in “Choose data correction actions,” you see the JOB_ID rule displayed. In the Cleanse Strategy column, select Similarity Match from the drop-down list. Click Next.

Similarity Match will match the erroneous value with the closest correct value in the domain and cleanse it appropriately using the Warehouse Builder match/merge operation.

Correct	Table	Description
<input checked="" type="checkbox"/>	EMPLOYEES	

Rule	Rule type	Action	Cleanse Strategy	Description
1 JOB_ID	Domain List	Cleanse	Similarity Match	

Note: The “Choose data correction actions” page is where you specify the actual action that the mapping applies to enforce your data rules.

9. Examine the Summary page, and click Finish.

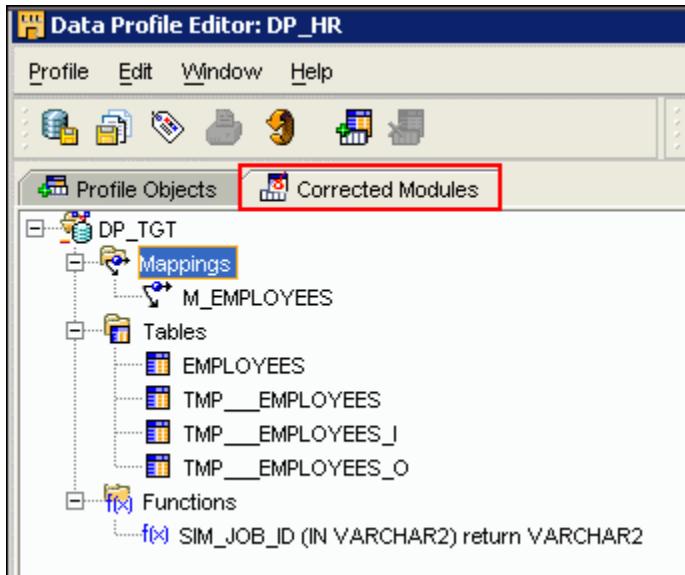
Correction definition is complete

When you click Finish, the wizard will create the object with the following settings:

- ❑ Target Module
DP_TGT
- ❑ Tables
TMP__EMPLOYEES
EMPLOYEES
TMP__EMPLOYEES_I
TMP__EMPLOYEES_O
- ❑ Maps
M_EMPLOYEES
- ❑ Functions
SIM_JOB_ID
Custom Functions

Now you can see the results of the generation. In the Data Profile Editor, click the Corrected Modules tab. You see the tables and the mapping.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)



Note: Some of the numbering in the tables will change depending on the number of times you created a correction map. Ensure that the SIM_JOB_ID function is present after generating the correction module.

- 6) Save your work.

Answer:

In the Data Profile Editor, click the Save All icon on the toolbar.

- 7) Close the Data Profile Editor.

Answer:

From the Profile menu, select Close.

Deploy and Run Correction Mappings

- 8) Launch the Control Center Manager.

Answer:

In the Design Center, select Control Center Manager from the Tools menu.

- 9) Register DP_TGT_LOCATION.

Answer:

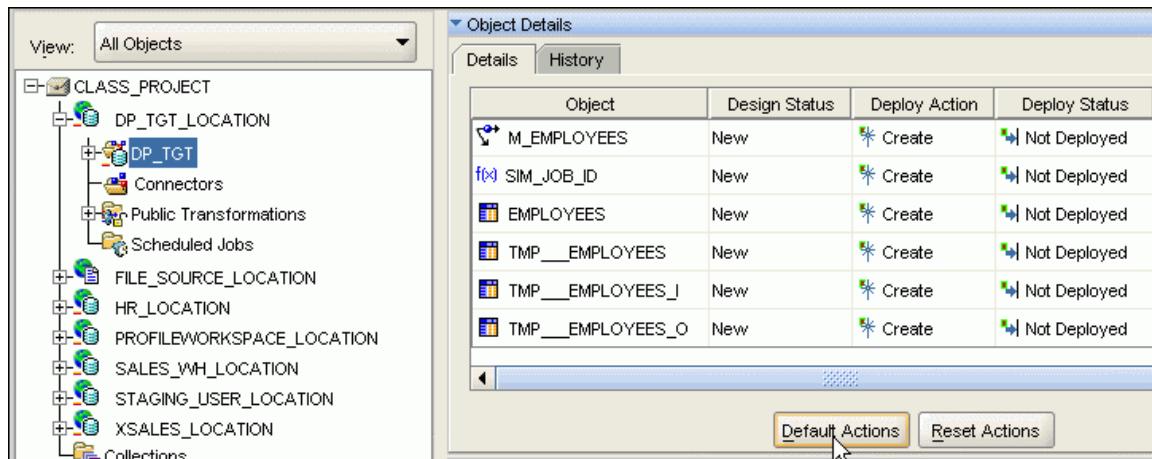
In the Center Manager, right-click DP_TGT_LOCATION, and select Register. Ensure that all values have been entered, and click Test Connection. If the connection is successful, click OK.

- 10) Deploy the mapping and its related objects.

Answer:

1. Expand DP_TGT_LOCATION > DP_TGT, and click Default Action. (Ensure that all objects are set to Create).

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)



2. On the toolbar, click Deploy.



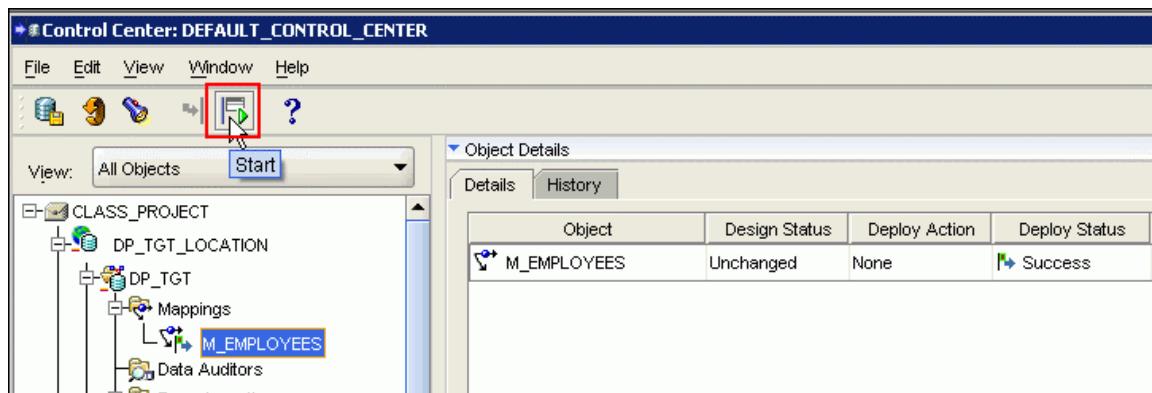
3. Monitor the progress of the deployment in the Control Center Jobs panel. When the deployment completes successfully, you are ready to execute the mapping.

Execute the Mapping

- 11) Execute the M_EMPLOYEES mapping.

Answer:

Expand DP_TGT > Mappings. Select M_EMPLOYEES, and click the Start icon on the toolbar.



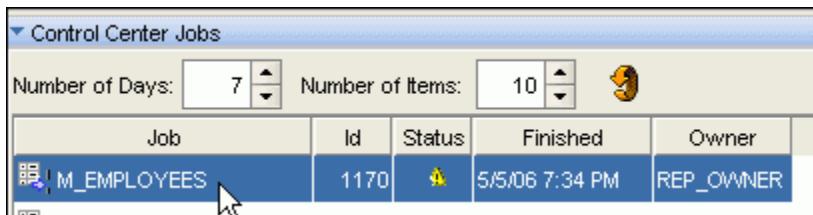
Note: Ignore the execution warnings, and examine the execution results.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

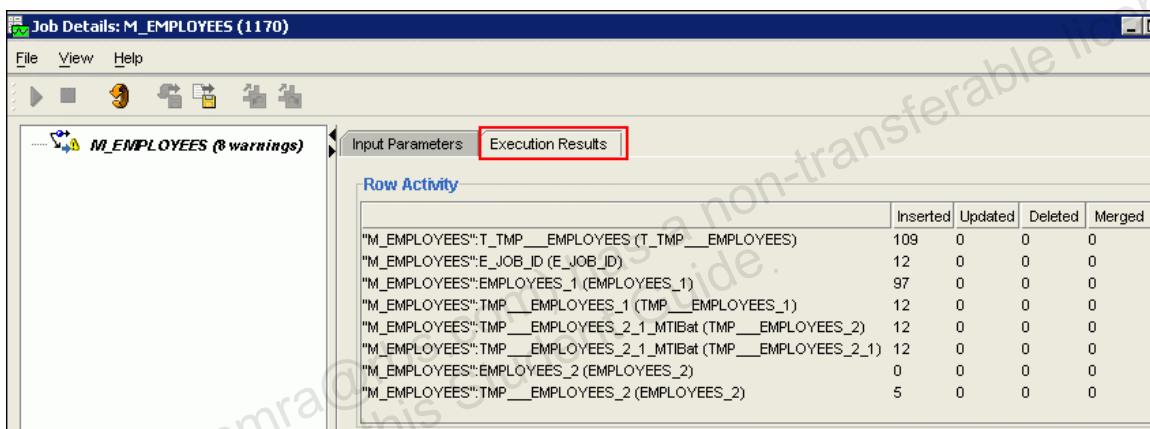
12) In the Job Details window, examine the results of execution.

Answer:

1. In the Control Center Jobs panel, select M_EMPLOYEES, and double-click in that row. The Job Details window appears.



2. In the Job Details window, click the Execution tab.



Note: Refer to the screenshot above:

T_TMP_EMPLOYEES: 109 rows inserted is the staging area receiving all selected rows from the EMPLOYEES table from the HR schema.

E_JOB_ID: 12 rows inserted that are not according to the domain.

EMPLOYEES_1: 97 rows is the first load into the final target of all initially correct rows.

EMPLOYEES_2: 5 rows is the first load into the final target of all cleansed according to similarity match rows.

13) Close the Job Details window and the Control Center Manager.

14) Check the result of your work. There should be no entries such as PU_CLIRK or SA_REPP.

Answer:

1. In the Project Explorer, expand CLASS_PROJECT > Databases > Oracle > DP_TGT > Tables. Right-click EMPLOYEES, and select Data.

Solutions for Practice 6-1: Deriving Data Rules and Creating Correction Mappings (continued)

The screenshot shows the Relational Data Viewer application window titled "Relational Data Viewer : EMPLOYEES". The menu bar includes "Object", "Window", and "Help". Below the menu is a toolbar with three buttons: "Execute Query" (highlighted in orange), "Get More", and "Where Clause...". The main area displays a table with the following data:

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY
43	121	Adam	Frapp	AFRIPP	650.123.2234	10-APR-97	PU_CLERK	8200
44	196	Alana	Walsh	AWALSH	650.507.9811	24-APR-98	SH_CLERK	3100
25	147	Alberto	Errazuriz	AERRAZUR	011.44.1344....	10-MAR-97	PU_CLERK	12000
36	115	Alexander	Khoo	AKHOO	515.127.4562	18-MAY-95	PU_CLERK	3100
86	103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-90	PU_CLERK	9000
41	185	Alexis	Bull	ABULL	650.509.2876	20-FEB-97	SH_CLERK	4100
32	158	Allan	McEwen	AMCEWEN	011.44.1345....	01-AUG-96	SA REP	9000
19	175	Alyssa	Hutton	AHUTTON	011.44.1644....	19-MAR-97	SA REP	8800
1	167	Amit	Banda	ABANDA	011.44.1346....	21-APR-00	SA REP	6200
42	187	Anthony	Cabrio	ACABRIO	650.509.4876	07-FEB-99	SH_CLERK	3000

- 15) Close the Relational Data Viewer.

Practice Solutions for Lesson 7

In this practice, you complete designing the partially created CLASS_PROJECT data warehouse project.

In Part 1 of the practice, you create the CHANNELS dimension using the wizard.

In Part 2 of the practice, you create the PRODUCTS dimension using the editor.

In Part 3 of the practice, you create the TIMES dimension using Time wizard.

In Part 4 of the practice, you create the SALES cube using the editor.

In Part 5 of the practice, you design the mapping to load the PRODUCTS dimension.

OWB automatically creates for you the mapping to load the TIMES dimension. To save time during this class, you run a script that creates the LOAD_SALES mapping to load the SALES cube. The guided instructions to create the LOAD_SALES cube mapping are in the appendix titled “Create the LOAD_SALES Mapping.”

Solutions for Practice 7-1: Create a Dimension Using the Wizard

In this practice, you create the CHANNELS dimension by using the Create Dimension Wizard.

- 1) Use the Connection Explorer to examine whether you have access to the XSALES source schema.

Answer:

1. In the Connection Explorer panel, expand Locations > Databases > Oracle.
2. Double-click XSALES_LOCATION. In the Edit Connection dialog box, all fields have values; enter XSALES in the password field (if not already entered), and click Test Connection. If the connection is successful, click OK.

Create the CHANNELS Dimension Using the Wizard

- 2) Using the wizard, create the CHANNELS dimension under the SALES_WH target module. Make note of the following details:
 - a) Name: CHANNELS
 - b) Storage Type: ROLAP: (Relational Storage)
 - c) For Dimension Attributes follow the details as shown below:

Name	Identifier	Data Type	Length	Descriptor
ID	Surrogate	NUMBER		
NAME		VARCHAR2	60	Short Description
SOURCE_ID	Business	VARCHAR2	40	

- d) Levels page: Create three levels as:

Name	Description
TOTAL	Total
CLASS	Class
CHANNEL	Channel

- e) Level attributes page: Specify the attributes for each level as follows:

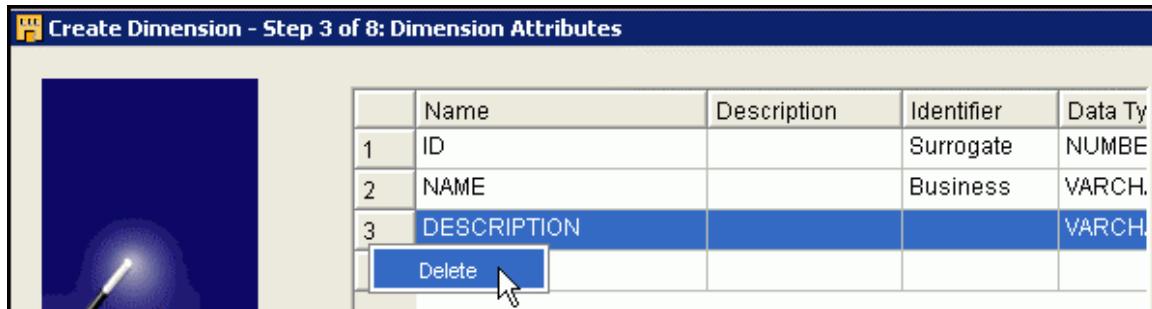
Level	Level Attributes
TOTAL	ID
	NAME
	SOURCE_ID

**Solutions for Practice 7-1: Create a Dimension Using the Wizard
(continued)**

CLASS	ID
	NAME
	SOURCE_ID
CHANNEL	ID
	NAME
	SOURCE_ID

Answer:

1. In the Design Center, in the Project Explorer, expand CLASS_PROJECT > Databases > Oracle > SALES_WH.
2. Right-click Dimensions, select New > Using Wizard. The Create Dimension Wizard is launched. Click Next on the Welcome page.
3. On the Name and Description page, enter the following:
 - a. Name: CHANNELS
 - b. Description: Channels DimensionClick Next.
4. On the Storage Type page, choose ROLAP: (Relational storage), and click Next.
5. On the Dimension Attributes page, you find three predefined columns: ID, NAME, and DESCRIPTION.
 - a. Delete the DESCRIPTION column. Right-click in the column on the left of DESCRIPTION, and select Delete.



- b. For the Name attribute, select empty value from the Identifier drop-down list, and change Length to 60.

**Solutions for Practice 7-1: Create a Dimension Using the Wizard
(continued)**

Create Dimension - Step 3 of 8: Dimension Attributes						
	Name	Description	Identifier	Data Type	Length	
1	ID		Surrogate	NUMBER		
2	NAME			VARCHAR2	60	

- c. Add a new attribute SOURCE_ID as shown in the screenshot, and click Next.

Create Dimension - Step 3 of 8: Dimension Attributes									
	Name	...	Identifier	Data Type	Length	Precision	Scale	...	Descriptor
1	ID		Surrogate	NUMBER	0	0			
2	NAME			VARCHAR2	60				Short description
3	SOURCE_ID		Business	VARCHAR2	40				

6. On the Levels page, create the three levels as follows:

Create Dimension - Step 4 of 8: Levels		
Specify the levels in the default hierarchy:		
	Name	Description
1	TOTAL	Total
2	CLASS	Class
3	CHANNEL	Channel

7. On the Level Attributes page, specify the attributes for each level.

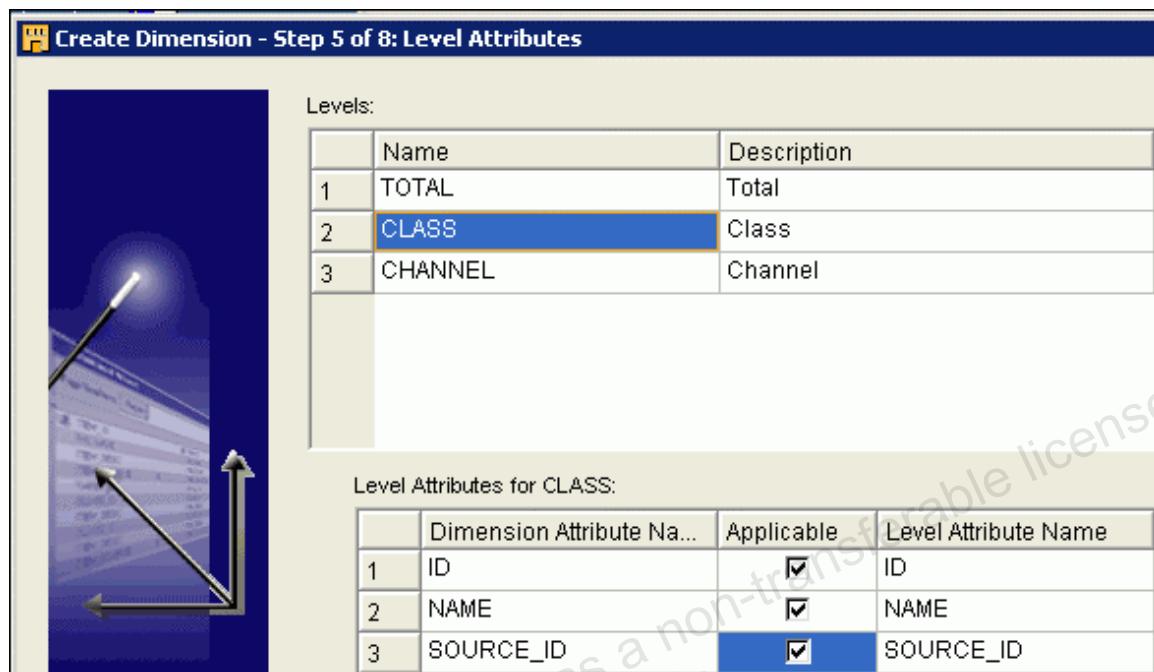
- a. For the TOTAL level, select the following attributes:

Create Dimension - Step 5 of 8: Level Attributes			
Levels:			
	Name	Description	
1	TOTAL	Total	
2	CLASS	Class	
3	CHANNEL	Channel	

Level Attributes for TOTAL:			
	Dimension Attribute Na...	Applicable	Level Attribute Name
1	ID	<input checked="" type="checkbox"/>	ID
2	NAME	<input checked="" type="checkbox"/>	NAME
3	SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID

Solutions for Practice 7-1: Create a Dimension Using the Wizard (continued)

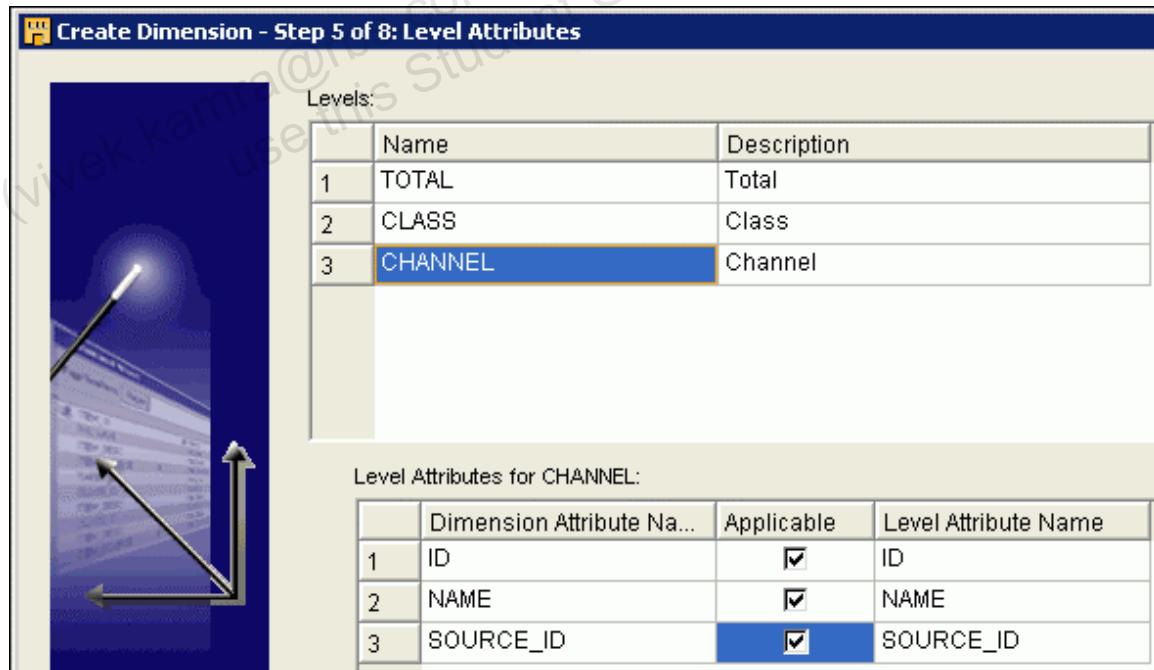
- b. For the CLASS level, select the following attributes:



	Name	Description
1	TOTAL	Total
2	CLASS	Class
3	CHANNEL	Channel

	Dimension Attribute Name	Applicable	Level Attribute Name
1	ID	<input checked="" type="checkbox"/>	ID
2	NAME	<input checked="" type="checkbox"/>	NAME
3	SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID

- c. For the CHANNEL level, select the following attributes:



	Name	Description
1	TOTAL	Total
2	CLASS	Class
3	CHANNEL	Channel

	Dimension Attribute Name	Applicable	Level Attribute Name
1	ID	<input checked="" type="checkbox"/>	ID
2	NAME	<input checked="" type="checkbox"/>	NAME
3	SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID

Note: For the lowest level, all the attributes are selected by default.

8. Click Next. The Slowly changing Dimension page appears. On the Slowly Changing Dimensions page, accept the default choice of **Type 1: Do not keep history**. Click Next.
9. Examine the “Pre-Create setting” page, and click Next.

Solutions for Practice 7-1: Create a Dimension Using the Wizard (continued)

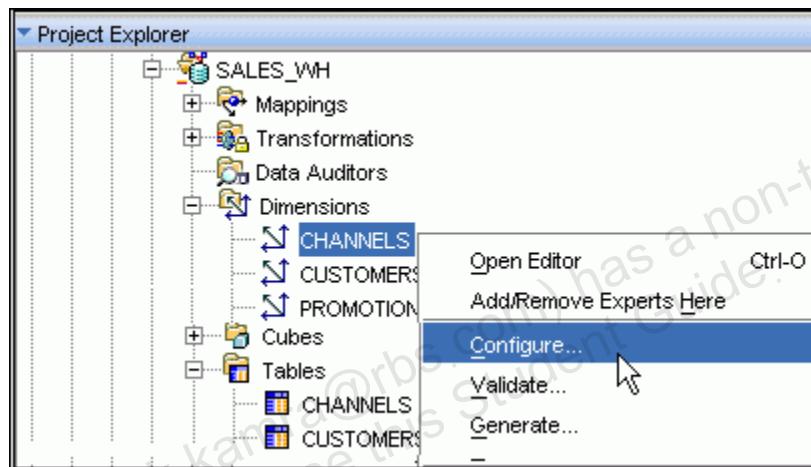
10. The Dimension Creation Progress implements the dimension, and the dimension is successfully created. Click Next.
11. Click Finish on the Summary page.

The CHANNELS dimension is created for you. You now configure one of the dimension properties.

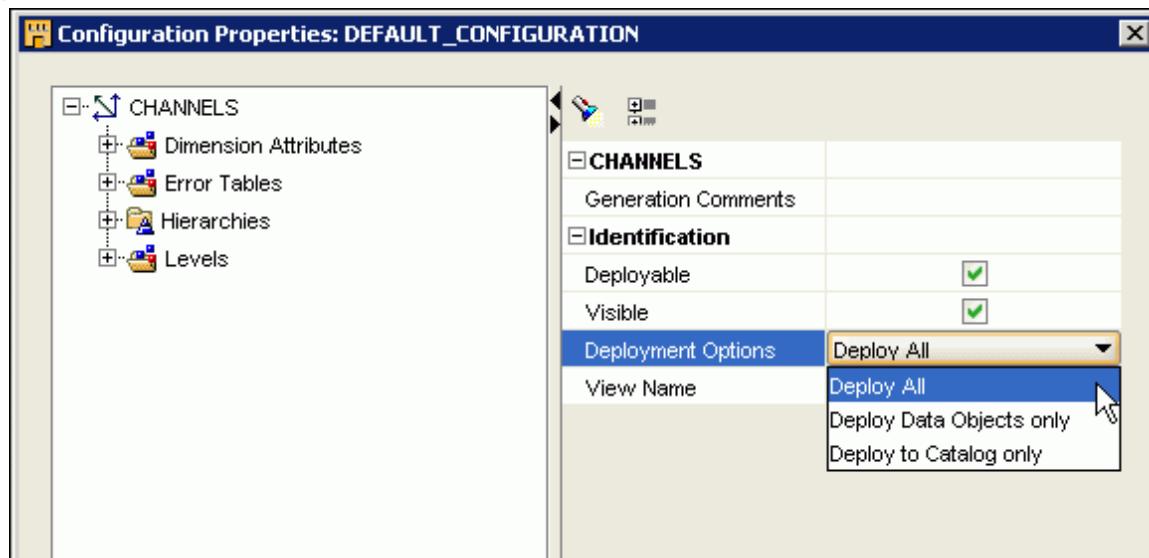
- 3) Assign **Deploy All** as the deployment option for the CHANNELS dimension.

Answer:

1. From the Project Explorer, expand SALES_WH > Dimension. Right-click CHANNELS, and select Configure. The Configuration Properties dialog box appears.



2. Select Deployment Options, and select Deploy All from the drop-down list. Click OK.



- 4) Save your work.

Answer:

**Solutions for Practice 7-1: Create a Dimension Using the Wizard
(continued)**

In the Design Center, click the Save All icon on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

Solutions for Practice 7-2: Create a Dimension Using the Editor

In this practice, you create the PRODUCTS dimension by using the editor.

- 1) Create the PRODUCTS dimension by using the editor. The dimension should be created under the SALES_WH target module and with the following details on the various tabs:

- a) Name tab:

Name: PRODUCTS

Description: Products Dimension

- b) Storage tab:

Accept default option ROLAP: (Relational Storage)

Implementation: Star

- c) Attributes tab:

Sequence: Select **PROD_DIM_SEQ** from the list.

Name	Identifier	Data Type	Length
ID	Surrogate	Number	
Name		Varchar2	60
Description		Varchar2	100
Source_Id	Business	Varchar2	40
Pack_Size		Varchar2	30
List_Price		Varchar2	10

- d) Levels tab:

Level	Description
TOTAL	Total
CATEGORY	Category
SUBCATEGORY	Sub Category
PRODUCT	Product

Select the following attributes for each level:

Level	Level Attributes

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

TOTAL	ID
	Name
	Description
	Source_Id
CATEGORY	ID
	Name
	Description
	Source_Id
SUBCATEGORY	ID
	Name
	Description
	Source_Id
PRODUCT	ID
	Name
	Description
	Source_Id
	Pack_Size
	List_Price

e) Hierarchies tab:

Hierarchy: **PROD_STD**

Levels for PROD_STD:

Level
TOTAL
CATEGORY

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

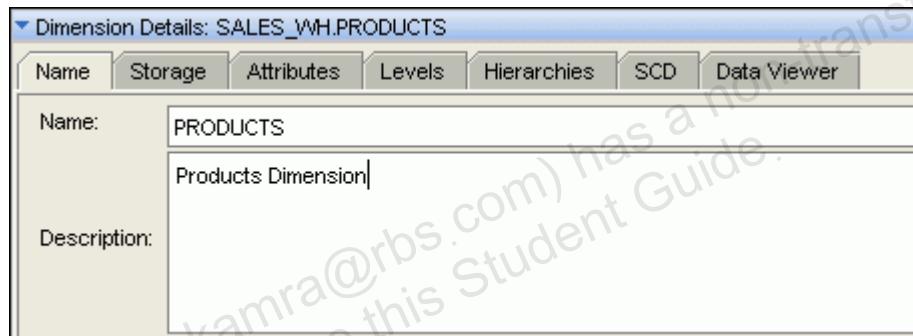
SUBCATEGORY
PRODUCT

- f) SCD tab:

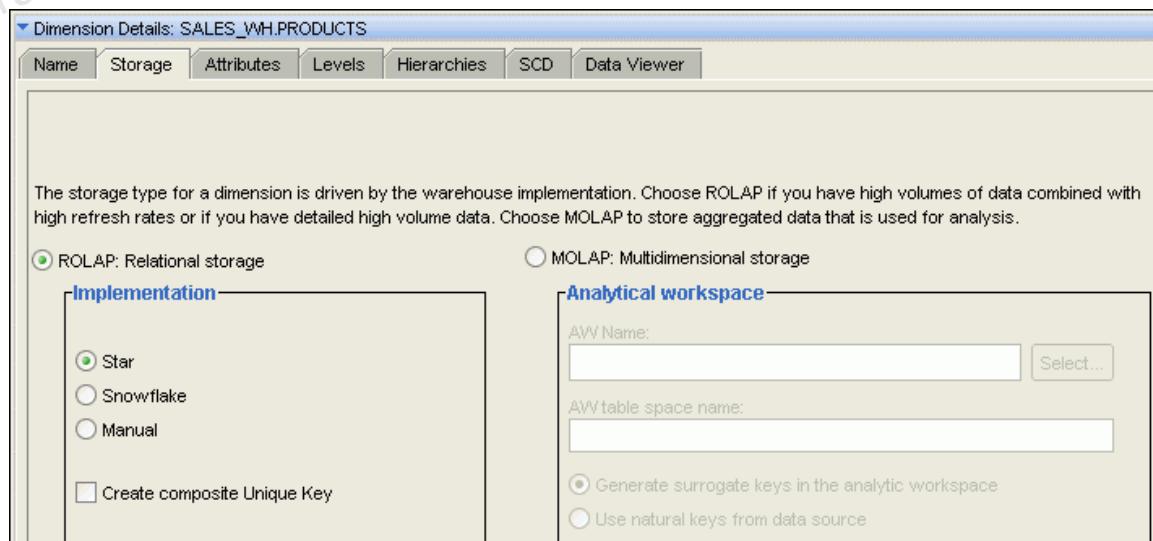
Accept the default selection, Type 1: Do not keep history.

Answer:

1. In the Project Explorer in the Design Center, expand CLASS_PROJECT > Databases > Oracle > SALES_WH. Right-click Dimensions, and select New > Using Editor. The Data Object Editor window appears.
2. In the Dimension Details window, ensure that the Name tabbed is displayed. In the Name field, change DIMENSION_1 to PRODUCTS. In the Description field, enter Products Dimension.



3. Click the Storage tab. Accept the default options, Relational: Relational data structures. For Implementation, select Star Implementation.



4. Click the Attributes tab. Click Select. In the Available Sequence dialog box, expand SALES_WH, and select PROD_DIM_SEQ. Click OK. The sequence populates the dimension key. Enter the following attributes as shown:

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

The screenshot shows the 'Dimension Details' dialog for the 'SALES_WH.PRODUCTS' dimension. The 'Attributes' tab is selected. A message at the top says 'Choose the sequence that will populate the Dimension and Surrogate Keys:' followed by a dropdown menu containing 'PROD_DIM_SEQ'. A 'Select...' button is highlighted with a mouse cursor. Below is a table of attributes:

	Name	Description	Identifier	Data Type	Length
1	ID		Surrogate	NUMBER	
2	NAME			VARCHAR2	60
3	DESCRIPTION			VARCHAR2	100
4	SOURCE_ID		Business	VARCHAR2	40
5	PACK_SIZE			VARCHAR2	30
6	LIST_PRICE			VARCHAR2	10

5. Click the Levels tab. The Levels tabbed page has two sections, Levels and Level Attributes for *<a selected level>*.

- a. In the Levels section, enter the following:

The screenshot shows the 'Dimension Details' dialog for the 'SALES_WH.PRODUCTS' dimension. The 'Levels' tab is selected. It displays a table of levels:

	Level	Description
1	TOTAL	Total
2	CATEGORY	Category
3	SUBCATEGORY	Sub Category
4	PRODUCT	Product

- b. In "Level Attributes for *<a selected level>*," specify the following:

Level attributes for TOTAL are as follows:

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

Name	Storage	Attributes	Levels	Hierarchies	SCD	Data Viewer
1	TOTAL		Description			
2	CATEGORY		Total			
3	SUBCATEGORY		Category			
4	PRODUCT		Sub Category			
			Product			

Level Attributes for TOTAL:

	Dimension Attribute Name	Applica...	Level Attribute Name
1	ID	<input checked="" type="checkbox"/>	ID
2	NAME	<input checked="" type="checkbox"/>	NAME
3	DESCRIPTION	<input checked="" type="checkbox"/>	DESCRIPTION
4	SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID
5	PACK_SIZE	<input type="checkbox"/>	
6	LIST_PRICE	<input type="checkbox"/>	

Level attributes for CATEGORY are as follows:

Dimension Details: SALES_WH.PRODUCTS					
Name	Storage	Attributes	Levels	Hierarchies	SCD
			Level		
1	TOTAL			Description	
2	CATEGORY			Total	
3	SUBCATEGORY			Category	
4	PRODUCT			Sub Category	
				Product	
Level Attributes for CATEGORY:					
	Dimension Attribute Name	Applica...	Level Attribute Name		
1	ID	<input checked="" type="checkbox"/>	ID		
2	NAME	<input checked="" type="checkbox"/>	NAME		
3	DESCRIPTION	<input checked="" type="checkbox"/>	DESCRIPTION		
4	SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID		
5	PACK_SIZE	<input type="checkbox"/>			
6	LIST_PRICE	<input type="checkbox"/>			

Level attributes for SUBCATEGORY are as follows:

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

Dimension Details: SALES_WH.PRODUCTS			
Name	Storage	Attributes	Levels
Hierarchies	SCD	Data Viewer	
Level		Description	
1	TOTAL	Total	
2	CATEGORY	Category	
3	SUBCATEGORY	Sub Category	
4	PRODUCT	Product	
Level Attributes for SUBCATEGORY:			
	Dimension Attribute Name	Applica...	Level Attribute Name
1	ID	<input checked="" type="checkbox"/>	ID
2	NAME	<input checked="" type="checkbox"/>	NAME
3	DESCRIPTION	<input checked="" type="checkbox"/>	DESCRIPTION
4	SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID
5	PACK_SIZE	<input type="checkbox"/>	
6	LIST_PRICE	<input type="checkbox"/>	

Level attributes for PRODUCT are as follows:

6. Click the Hierarchies tab. Specify the following details:

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

Dimension Details: SALES_WH.PRODUCTS

Hierarchy		Description	Default
1	PROD_STD		<input checked="" type="radio"/>
			<input type="radio"/>

Levels for PROD_STD:		
	Level	Skip to Level
1	TOTAL	
2	CATEGORY	
3	SUBCATEGORY	
4	PRODUCT	

7. Click the SCD tab. Accept the default selection, Type 1: Do not keep history.
- 2) Create the dimension table, and bind it to the repository. (Use Auto Bind.)

Answer:

1. On the canvas, right-click the PRODUCTS dimension, and select Auto Bind.

Canvas

Relational Dimensional **x** Business Definition

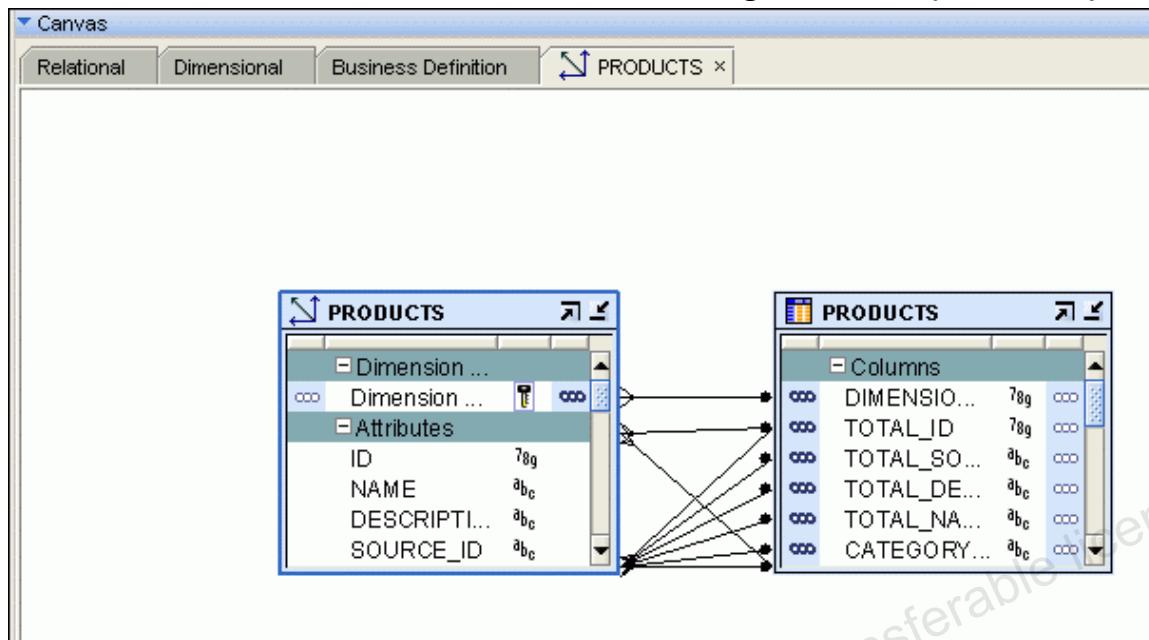
PRODUCTS

ID NAME DESCRIPTI... SOURCE_ID PACK_SIZE LIST_PRICE

Auto Bind

2. After you complete the binding process, this is what you see:

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

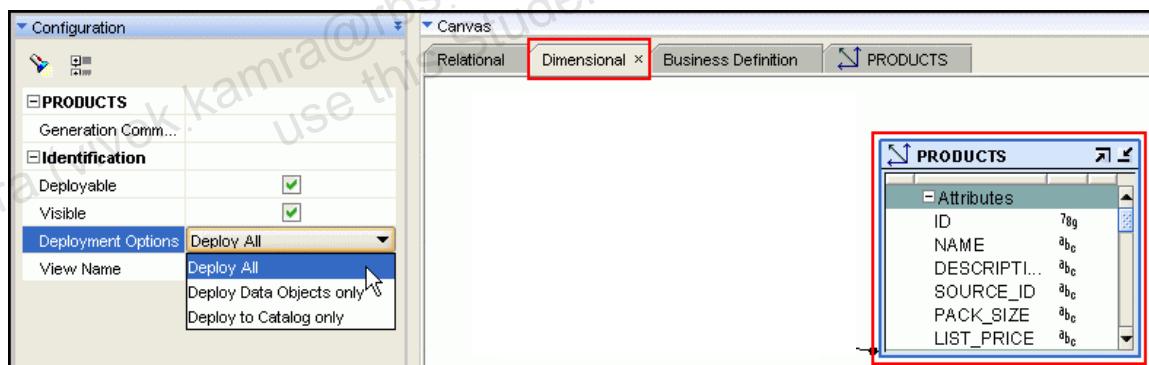


You have now created the PRODUCTS dimension.

- 3) Select **Deploy All** as Deployment Options for the PRODUCTS dimension.

Answer:

1. On the canvas, click the Dimensional tab, and select the PRODUCTS dimension.



2. From the Configuration palette, select Deployment Options. From the drop-down list, select Deploy All.

- 4) Save your work, and close the Data Object Editor.

Answer:

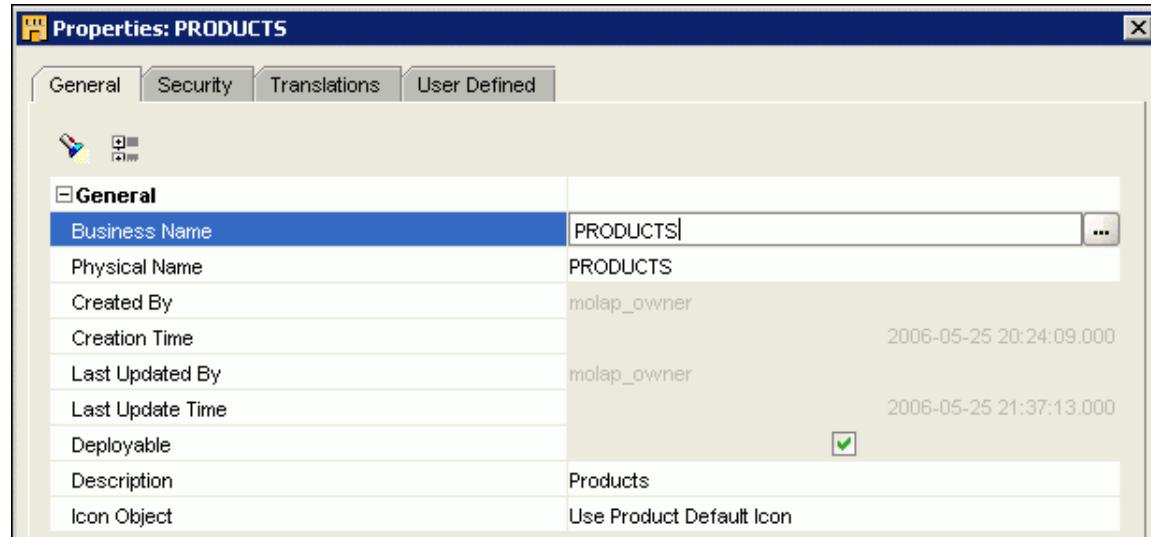
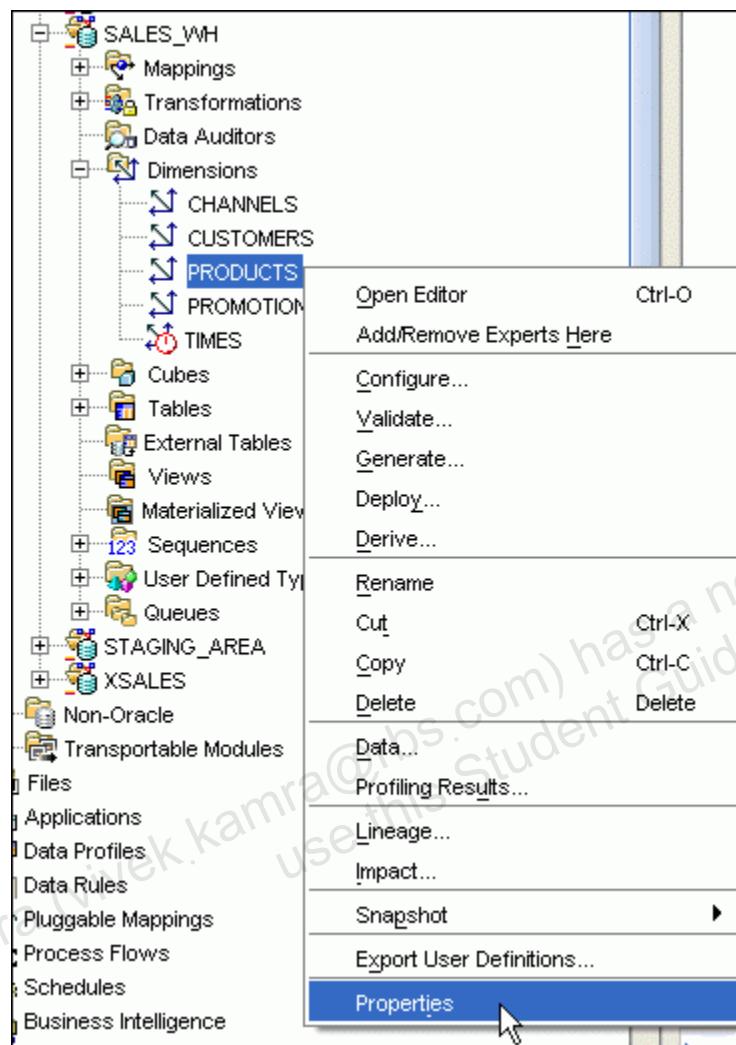
In the Data Object Editor, click Save All on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

- 5) Set the business name of the PRODUCTS dimension.

Answer:

Solutions for Practice 7-2: Create a Dimension Using the Editor (continued)

In the Project Explorer, expand SALES_WH > Dimensions. Right-click PRODUCTS, and select Properties. In the Properties window, enter Products in the Business Name field. Click OK.



Solutions for Practice 7-3: Create a TIMES Dimension Using the Time Wizard

In this practice, you create the TIMES dimension by using the Time Wizard.

- 1) Create the TIMES dimension by using the Time Wizard. Create this TIMES dimension under the SALES_WH module with the following details:

- a) Name and Description page:

Name: TIMES

Description: Times Dimension

- b) Storage Type page:

Accept the default, ROLAP: Relational Storage.

- c) Data Generation page:

Start year: 2001

Number of years: 1

- d) Levels page:

Normal Hierarchy

Levels in the Normal Hierarchy

Calendar Year

Calendar Quarter

Calendar Month

Answer:

1. In the Project Explorer in the Design Center, expand CLASS_PROJECT > Databases > Oracle > SALES_WH. Right-click Dimensions, and select New > Using Time Wizard. The Create Time Dimension Wizard is launched. Click Next on the Welcome page.

2. On the Name and Description page, enter the following:

Name: TIMES

Description: Times Dimension

3. Click Next.

4. On the Storage Type page, accept the default ROLAP: Relational Storage, and click Next. The Data Generation page appears.

5. Enter the following information:

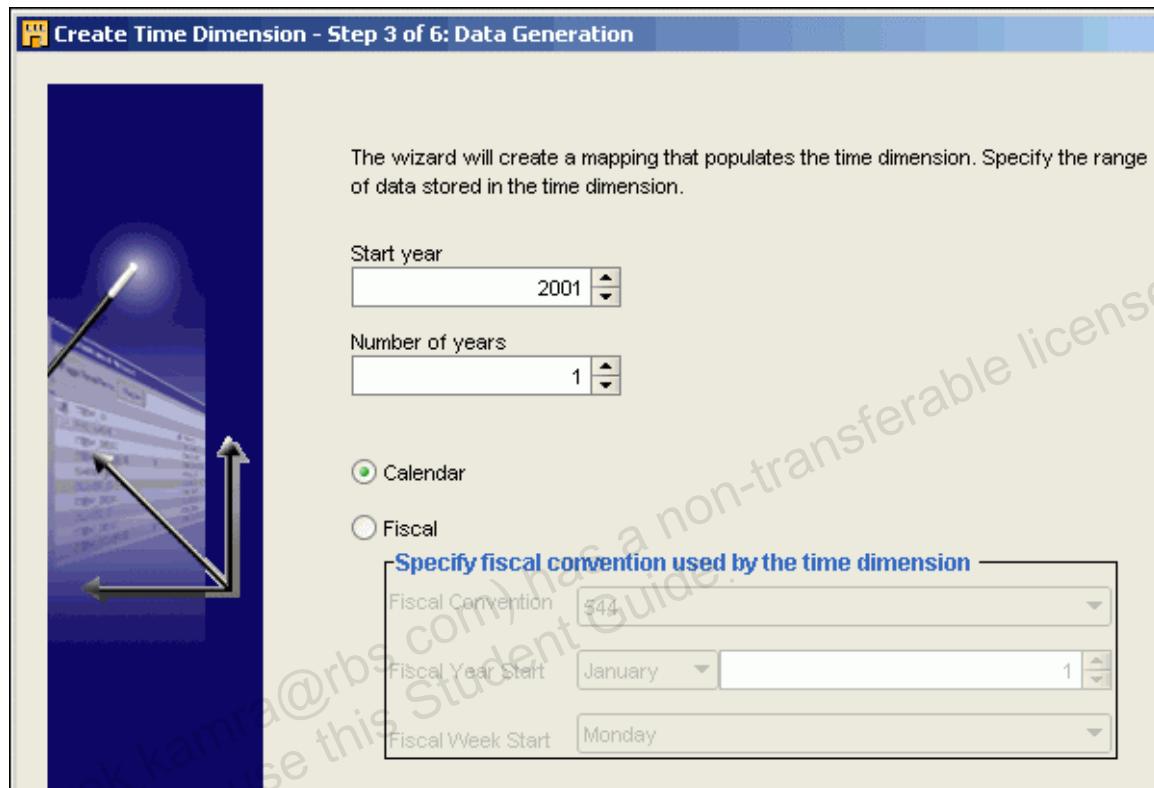
- a. Start Year: 2001

- b. Number of years: 1

Ensure that the Calendar option is selected.

Solutions for Practice 7-3: Create a TIMES Dimension Using the Time Wizard (continued)

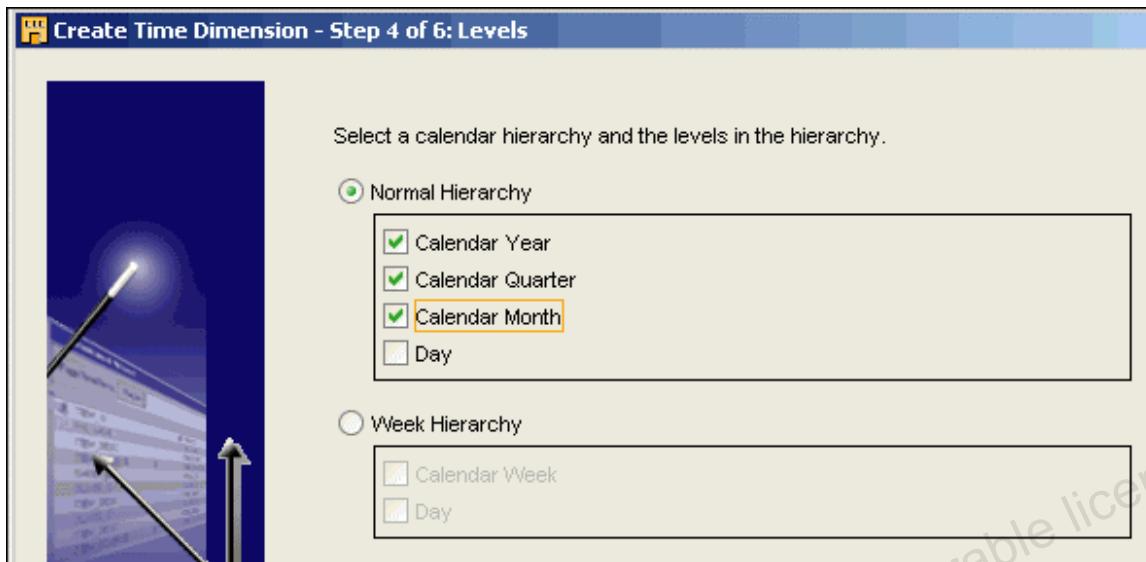
Note: On the Data Generation page, specify the range of time data that is required for your warehouse. This information is used to generate a mapping that will populate the time dimension. Within this mapping, the dates you enter are added as parameters, enabling you to rerun this mapping with dates at a later stage.



Click Next.

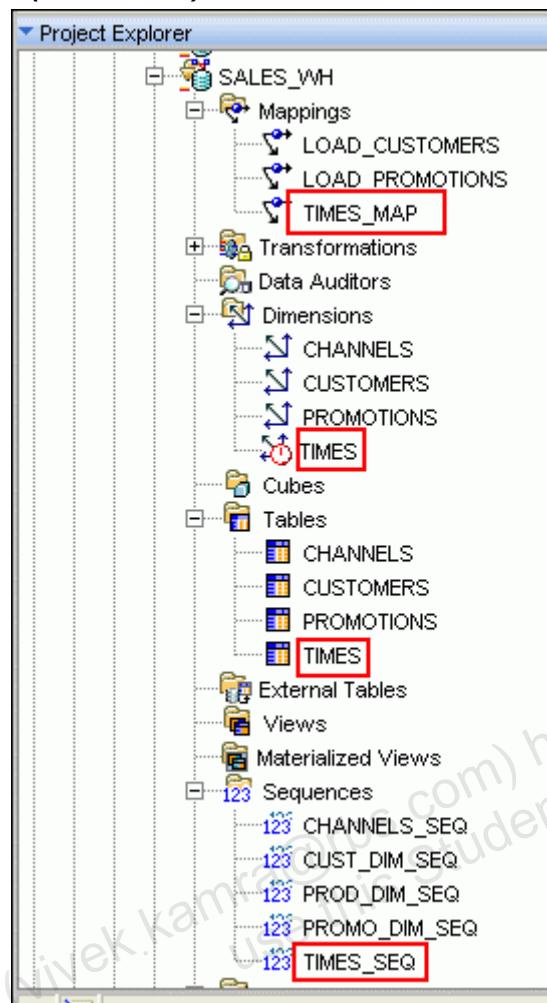
6. On the Levels page, select as shown in the screenshot, and click Next.

Solutions for Practice 7-3: Create a TIMES Dimension Using the Time Wizard (continued)



7. On the Pre Create Settings page, examine the details, and click Next. A Progress bar shows the progress as the wizard creates the objects.
8. On successful completion, click Next. On the Summary page, examine the details again, and click Finish.
9. In the Project Explorer, observe that the wizard has generated four objects necessary for a fully functional TIME dimension:
 - a. Expand CLASS_PROJECT > Databases > Oracle > SALES_WH > Dimension. You see the TIMES dimension.
 - b. Expand CLASS_PROJECT > Databases > Oracle > SALES_WH > Sequences. You see the TIMES_SEQ sequence that populates the surrogate ID of the time dimension levels.
 - c. Expand CLASS_PROJECT > Databases > Oracle > SALES_WH > Tables. The TIMES table to support the relational implementation of the time dimension that will physically store the time data.
 - d. Expand CLASS_PROJECT > Databases > Oracle > SALES_WH > Mappings. The TIMES_MAP mapping to populate the time dimension.

Solutions for Practice 7-3: Create a TIMES Dimension Using the Time Wizard (continued)

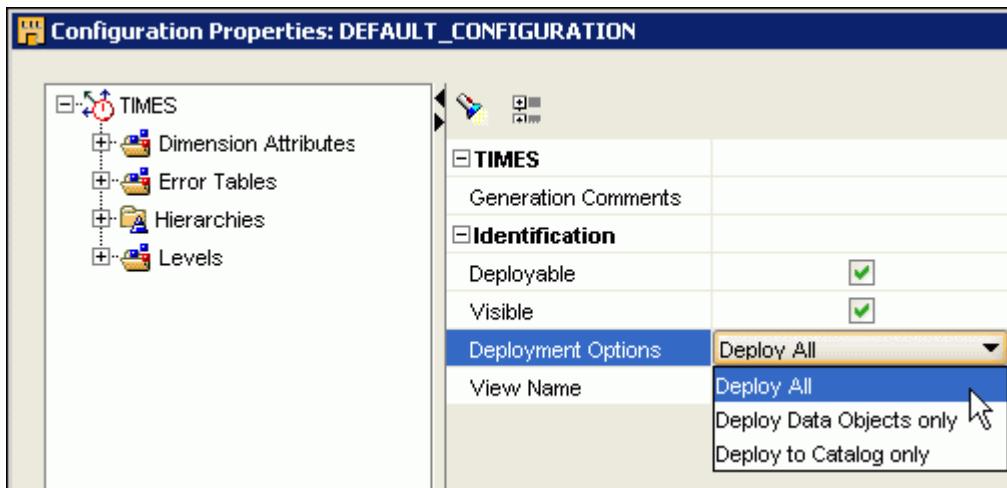


- 2) Select **Deploy All** as Deployment Options for the TIMES dimension.

Answer:

1. In the Project Explorer, expand SALES_WH > Dimension. Right-click TIMES, and select Configure. The Configuration Properties dialog box is displayed.
2. In the Configuration Properties dialog box, select Deployment Options from the right, and select Deploy All from the drop-down list. Click OK.

Solutions for Practice 7-3: Create a TIMES Dimension Using the Time Wizard (continued)



- 3) Save your work.

Answer:

In the Design Center, click the Save All icon on the toolbar. Click Yes in the Warehouse Builder Warning dialog box.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 7-4: Create a Cube Using the Editor

In this practice, you create a cube by using the editor.

Create the SALES Cube

- 1) Using the editor, create the sales cube with the following details:

- a) Name tab:

Name: SALES

- b) Storage tab:

ROLAP: Relational Storage

Select Create Bitmap Indexes.

- c) Dimensions tab:

DIMENSION	LEVEL
TIMES	CALENDAR_MONTH
PRODUCTS	PRODUCT
CHANNELS	CHANNEL
CUSTOMERS	CITY
PROMOTIONS	SUBCATEGORY

- d) Measures tab:

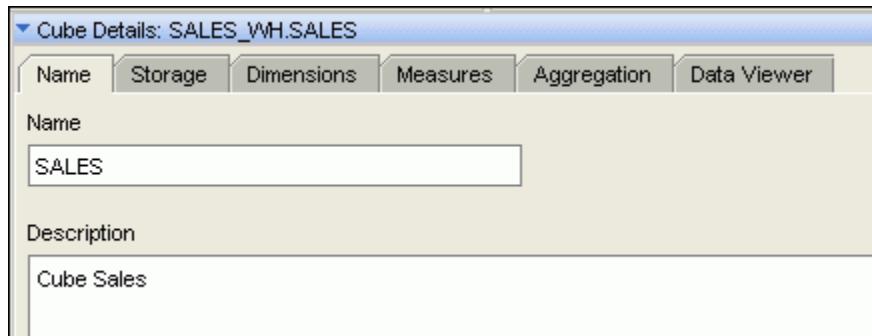
Name	Description	Data Type	Precision	Scale
Amount	Sales Amount	Number	10	2
Quantity	Sales Quantity	Number		
Cost	Sales Cost	Number	10	2

- e) Aggregation tab

Answer:

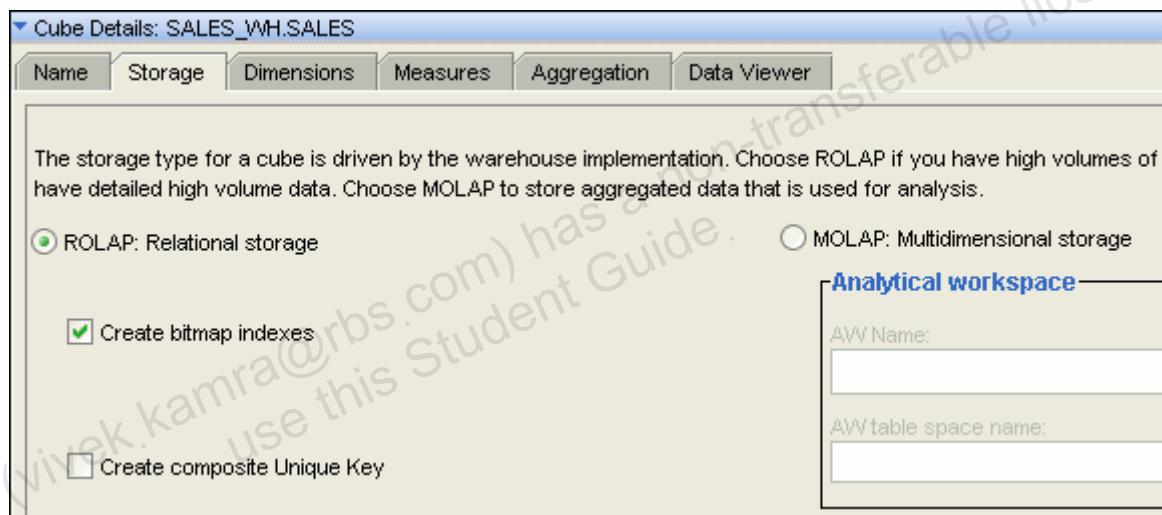
1. In the Project Explorer in the Design Center, expand CLASS_PROJECT > Databases > Oracle > SALES_WH. Right-click Cubes, and select New > Using Editor. The Data Object Editor window is displayed.
2. In the Cube Details window, ensure that the Name tabbed page is displayed. In the Name field, change CUBE_1 to SALES. In the Description field, enter Sales Cube.

Solutions for Practice 7-4: Create a Cube Using the Editor (continued)



3. Click the Storage tab. Accept the default selection:

- ROLAP: Relational data structures. Using this option, the cube definition and its data are stored in the relational form in the database.
- Ensure that the Create Bitmap Indexes option is selected.



4. Click the Dimensions tab. Select the dimension from the drop-down list in the dimension column and corresponding levels in the Level column. Level displays all the levels for that dimension.

Solutions for Practice 7-4: Create a Cube Using the Editor (continued)

Cube Details: SALES_WH.SALES				
Name	Storage	Dimensions	Measures	Aggregation
	Dimension	Level	Role	
1	TIMES	CALENDAR_MONTH		
2	PRODUCTS	PRODUCT		
3	CHANNELS	CHANNEL		
4	CUSTOMERS	CITY		
5	PROMOTIONS	SUBCATEGORY		

5. Click the Measures tab, and enter the information as in the screenshot:

Cube Details: SALES_WH.SALES							
Name	Storage	Dimensions	Measures	Aggregation	Data Viewer		
	Name	Description	Data Type	Length	Precision	Scale	Second...
1	AMOUNT	Sales Amount	NUMBER	10	2		
2	QUANTITY	Sales Quantity	NUMBER	0	0		
3	COST	Sales Cost	NUMBER	10	2		

6. Click the Aggregation tab. Accept the default Aggregation function SUM for all the dimensions. This is used to aggregate the cube data.

You can verify this by clicking each dimension and checking the cube aggregation method. In the following screenshots, you see the TIMES dimension selected and the cube aggregation method is SUM. Similarly, when you select the CHANNELS dimension, the cube aggregation method is SUM. Try this with the CUSTOMERS and PROMOTIONS dimensions.

Solutions for Practice 7-4: Create a Cube Using the Editor (continued)

The screenshot displays two instances of the Oracle Warehouse Builder 'Cube Details' interface for a cube named 'SALES_WH.SALES'.
The top instance shows the 'Dimensions' tab selected. A red box highlights the 'Cube Aggregation Method' dropdown, which is set to 'SUM'. Another red box highlights the first row of the 'Summary Strategy for cube SALES' table, which lists 'TIME' as dimension 1.
The bottom instance also shows the 'Dimensions' tab selected. Its 'Cube Aggregation Method' dropdown is also set to 'SUM'. The same red box highlights the first row of its 'Summary Strategy for cube SALES' table, listing 'TIME' as dimension 1.

	Dimension	Role	Pre Compute
1	TIMES		
2	PRODUCTS		
3	CHANNELS		
4	CUSTOMERS		
5	PROMOTIONS		

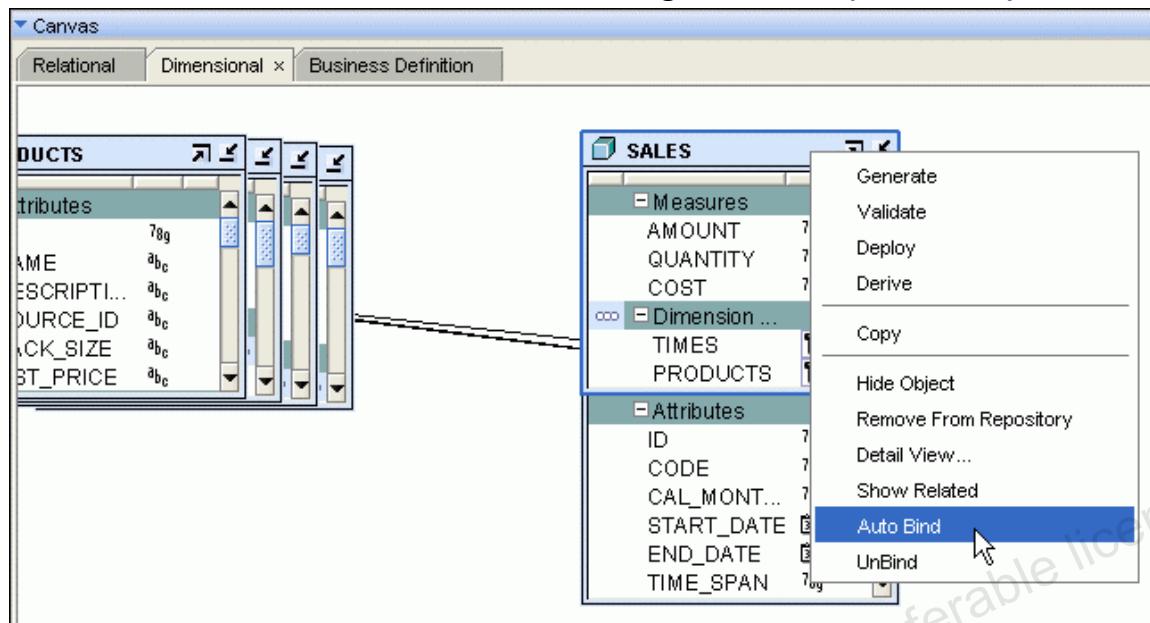
After defining the cube structure, you now specify the details about the database tables or views that store the cube data.

- 2) Create and bind the table to the repository.

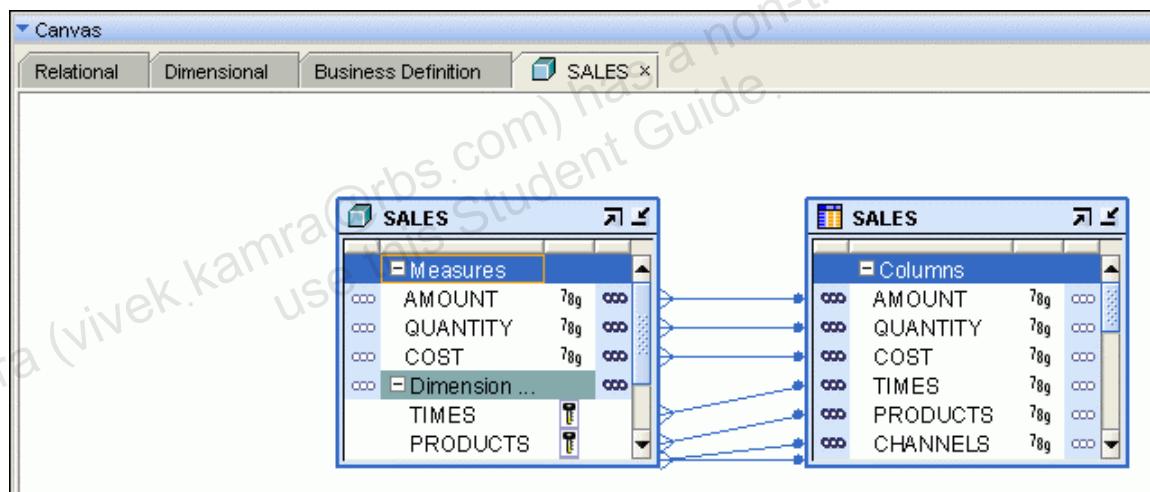
Answer:

On the canvas, in the Data Object Editor, right-click the SALES cube, and select Auto Bind. When you perform autobinding, Warehouse Builder automatically maps the measures and dimension references of the cube to the database columns that store their data.

Solutions for Practice 7-4: Create a Cube Using the Editor (continued)



After performing autobinding, this is what you see on the canvas.



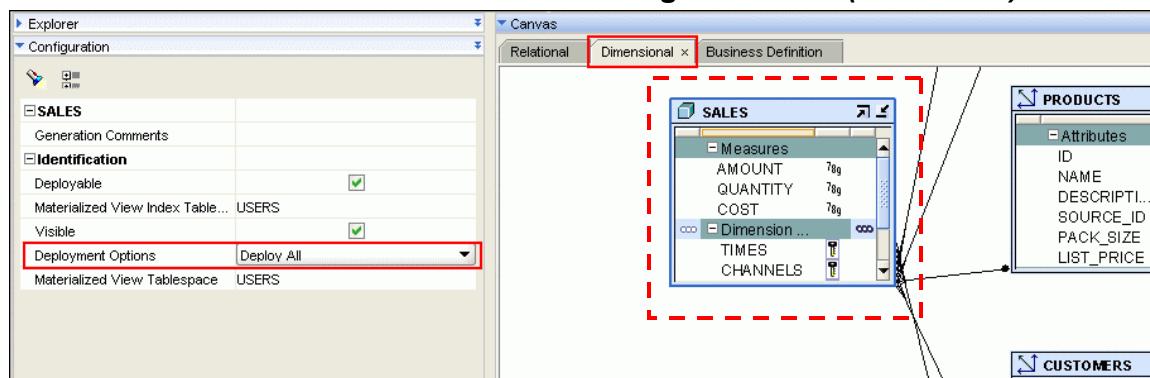
Maximize the SALES table operator and the SALES cube operator if you are not able to see the results clearly.

- 3) Configure Deployment Options for the SALES cube to **Deploy All**.

Answer:

In the Data Object Editor, select the SALES cube operator. In the Configuration palette, from the Deployment Options drop-down list, select **Deploy All**.

Solutions for Practice 7-4: Create a Cube Using the Editor (continued)



- Save your work, and close the Data Object Editor.

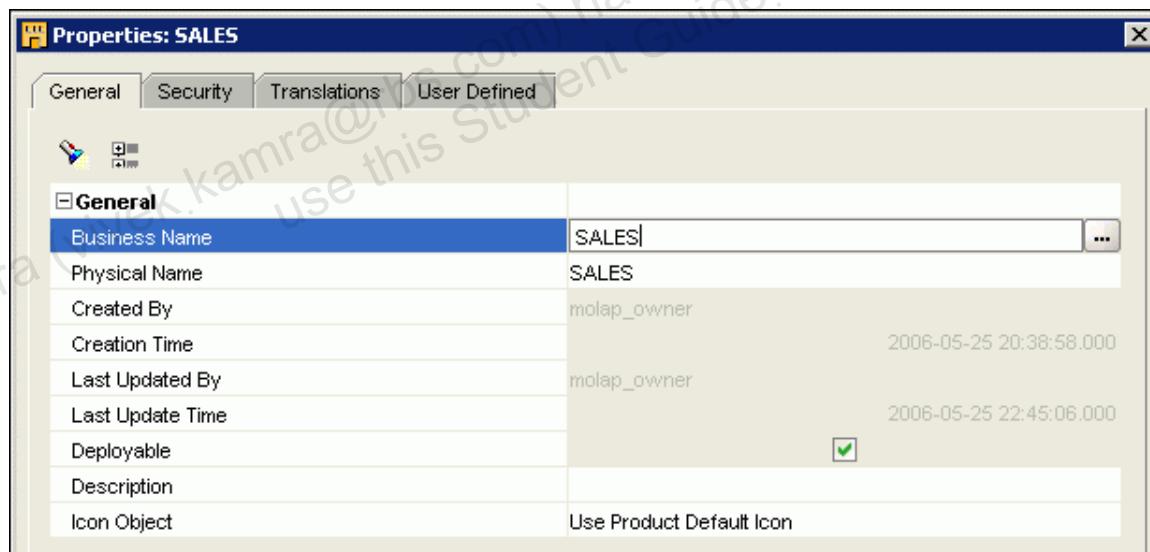
Answer:

Click the Save All icon on the toolbar. Click Yes, in the Warehouse Builder Warning dialog box. Close the Data Object Editor.

- Set Business Name for the SALES cube to SALES.

Answer:

In the Project Explorer, expand SALES_AW > Cubes. Right-click SALES, and select Properties. In the Properties window, in the Business Name field, enter SALES.

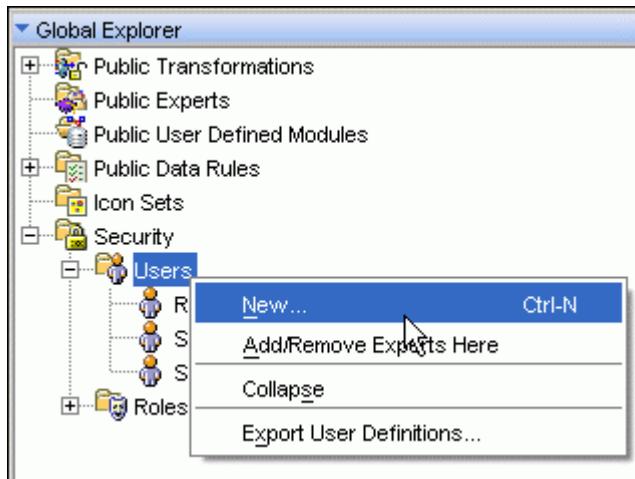


Solutions for Practice 7-5: Design ETL Mappings

In this practice, you design a mapping to load a dimension. Before you start creating a mapping, you register the SALES_WH database user as the target user for this class.

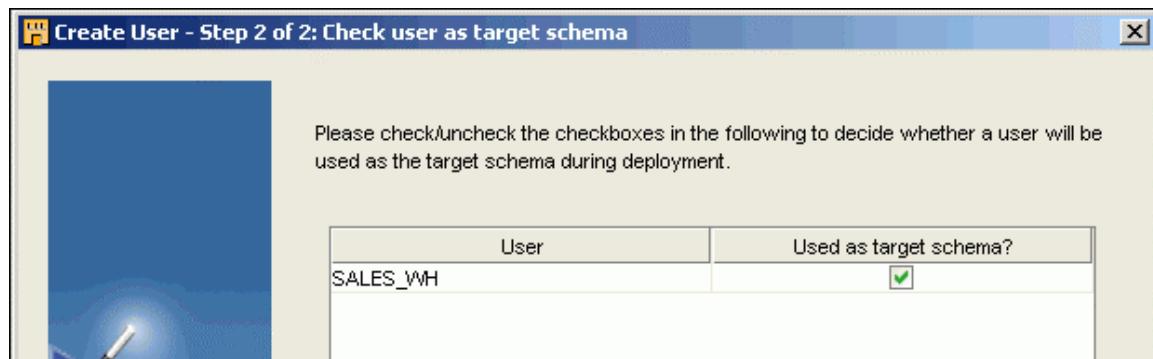
Registering a Database User as a Target User

- 1) Register the SALES_WH user as a target user. Use the Security node in the Global Explorer panel.



Answer:

1. In the Design Center, in the Global Explorer window, expand **Security**, right-click **Users**, and select **New**. The Create User Wizard is launched. Click **Next** on the Welcome page.
2. On the “Select DB user to register” page, from the Available DB Users list, select **SALES_WH** and shuttle it from the Available DB users list, and click **Next**.
3. On the “Check user as target schema” page, the SALES_WH user is selected to be used as the target schema. Click **Next**.

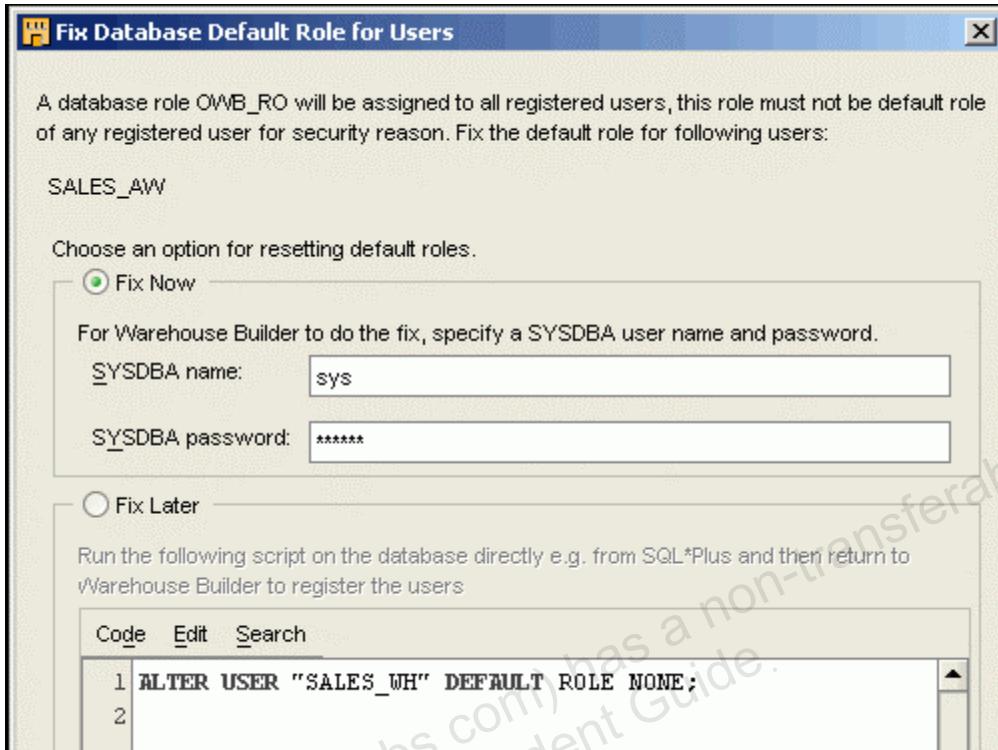


4. In the Target User Password dialog box, reenter the password as SALES_WH for the SALES_WH user. Click **OK**.
5. On the Summary page, review the summary, and click **Finish**. The Fix Database Default Role dialog box is displayed. In the Fix Database Default Role for Users dialog box, select **Fix Now**, and enter SYSDBA user information:
 - a. SYSDBA name: sys

Solutions for Practice 7-5: Design ETL Mappings (continued)

- b. SYSDBA password: oracle

Click OK.



The Register Users Progress dialog box displays the progress of the user creation. The target user definition is complete.

Note: For security reasons, OWB does not allow you to register database users with default roles in the database set to ALL. Warehouse Builder gives you options for changing the default setting. You correct the role by selecting Fix Now. Selecting the Fix Now option registers the SALES_AW user and assigns a database role OWB_<repository name> to the user.

- 2) Save your work.

Answer:

1. On the Design Center toolbar, click **Save All**.
- 3) Use the Connection Explorer to examine whether you have access to the SALES_WH schema.

Answer:

1. In the Connection Explorer panel, expand Locations > Databases > Oracle.
2. Double-click SALES_WH_LOCATION. In the Edit Connection dialog box, all fields have values; enter SALES_WH in the password field (if not already entered). Click Test Connection. If the connection is successful, click OK.

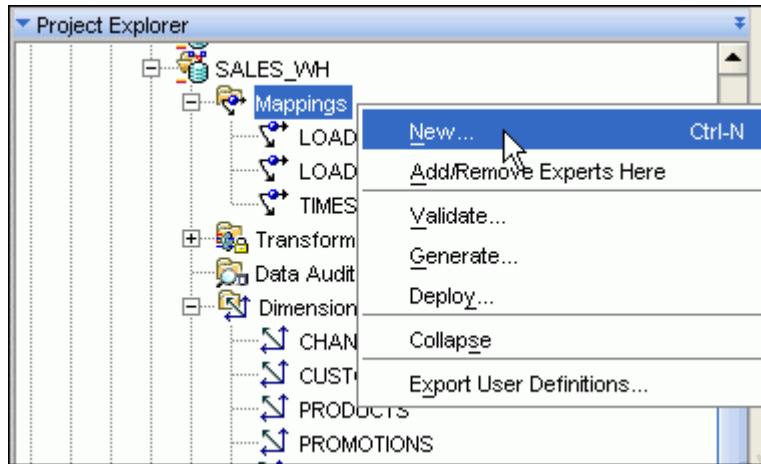
Design the LOAD_PRODUCTS Mapping to Load the PRODUCT Dimension

Solutions for Practice 7-5: Design ETL Mappings (continued)

- 4) Create a mapping named LOAD_PRODUCTS.

Answer:

1. In the Project Explorer, in the Design Center, expand CLASS_PROJECT > Databases > Oracle > SALES_WH. Right-click the Mappings node, and select New. The Create Mapping window appears.



2. In the Create Mapping window, enter LOAD_PRODUCTS as the name of the mapping. Click OK. The Mapping Editor appears.
- 5) Drag the following tables from the XSALES module to the canvas:
 - a) XSALES.CATEGORIES
 - b) XSALES.PRODUCTS

Answer:

1. In the Explorer panel, ensure that the **Available Objects** tabbed page is displayed. Expand Databases > Oracle > XSALES > Tables. Drag the CATEGORIES table to the canvas.
 - a. Drag the CATEGORIES table to the canvas.
 - b. Drag the PRODUCTS table to the canvas.

Solutions for Practice 7-5: Design ETL Mappings (continued)

The screenshot shows the Oracle Warehouse Builder interface. The left pane, titled 'Explorer', displays a tree view of database objects. Under 'Databases' > 'Oracle' > 'XSALES' > 'Tables', several tables are listed: ADDRESSES, CATEGORIES, CHANNELS, CITIES, COUNTRIES, CUSTOMERS, ORDERS, ORDER_ITEMS, PRODUCTS, PROMOTIONS, PROMO_CATEGORIES, PROMO_SUBCATEGORIES, and REGIONS. The 'PRODUCTS' table is selected and highlighted in blue. The bottom of the Explorer pane has two tabs: 'Available Objects' (which is redboxed) and 'Selected Objects'. The right pane, titled 'Mapping', contains two tables: 'PRODUCTS' and 'CATEGORIES'. Both tables have their first column ('INOUTG...') set to 'IN'. The 'PRODUCTS' table has columns: IDENTIFI..., SUBCAT..., NAME, DESCRI..., WEIGHT..., and UNIT_OF... with data types 78g, 3b, 3b, 3b, 3b, and 3b respectively. The 'CATEGORIES' table has columns: ID, CATEG..., NAME, DESCRI..., USER_C..., and DATE_C... with data types 78g, 3b, 3b, 3b, 3b, and 31 respectively.

- 6) Change the PRODUCTS table to PRODUCTS_IN.

Answer:

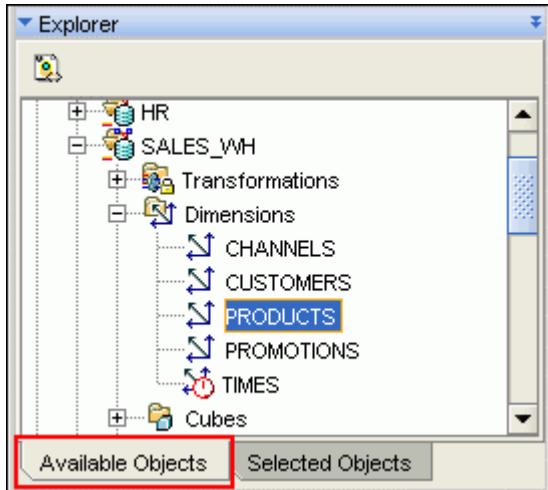
1. On the canvas, right-click the header area of the PRODUCTS table. Click Open Details.
2. In the Table Editor, ensure that the Name tabbed page is displayed. In the Name field, change PRODUCTS to PRODUCTS_IN. Click OK.

- 7) Drag the PRODUCTS dimension to the canvas from the SALES_WH module.

Answer:

1. In the Explorer panel, collapse the XSALES module, and expand SALES_WH > Dimension.

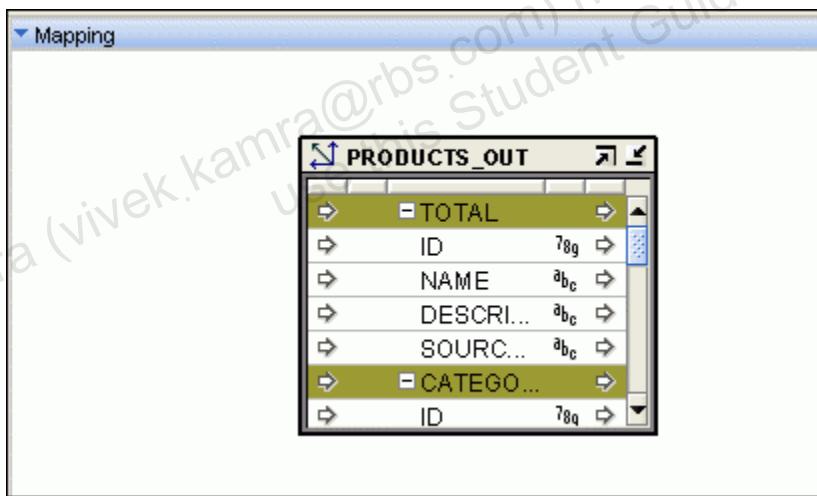
Solutions for Practice 7-5: Design ETL Mappings (continued)



2. Drag the PRODUCTS dimension to the canvas.
- 8) Change the PRODUCTS dimension to PRODUCTS_OUT.

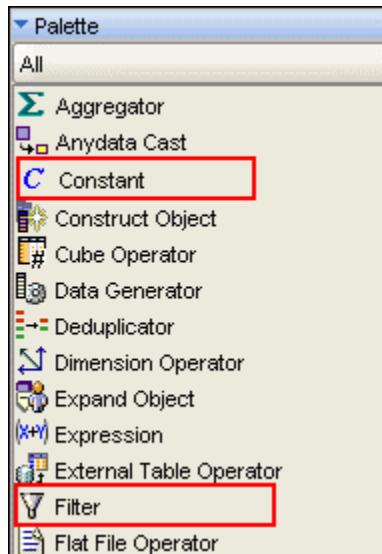
Answer:

Right-click the header area of the PRODUCTS dimension. Click Open Details. In the Table Editor, in the Name field, change PRODUCTS to PRODUCTS_OUT, and click OK.



- 9) From the Palette, drag the following three operators to the canvas:
 - a) CONSTANT
 - b) FILTER
 - c) FILTER
- Answer:**
1. From the Palette, drag the Constant operator to the canvas.
 2. From the Palette, drag the Filter operator to the canvas. Repeat this step to get the second filter operator onto the canvas.

Solutions for Practice 7-5: Design ETL Mappings (continued)



10) Rename the CONSTANT operator to TOTALS.

Answer:

1. Right-click the CONSTANT operator, and select Open Details.
2. In the Constant Editor, ensure that the Name tabbed page is displayed. In the Name field, change CONSTANT to TOTALS.

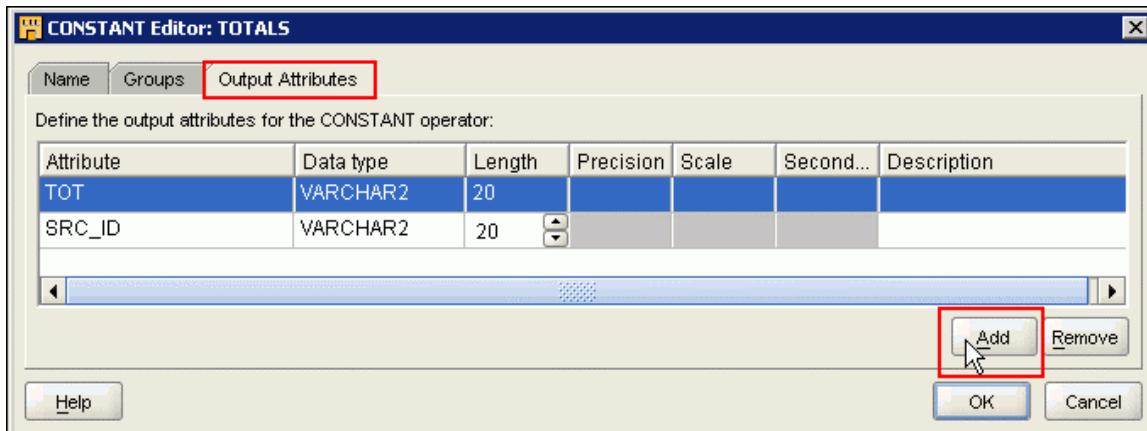
11) To the TOTALS constant operator, add the following two attributes with the following details:

Name	Data Type	Length
TOT	VARCHAR2	20
SRC_ID	VARCHAR2	20

Answer:

1. In the Constant Editor, click the Output Attributes tab.
2. At the lower-right corner, click Add. Change OUTPUT1 to TOT. Change the data type to VARCHAR2 and length to 20.
3. At the lower-right corner, click Add. Change OUTPUT1 to SRC_ID. Change the data type to VARCHAR2 and length to 20, and click OK.

Solutions for Practice 7-5: Design ETL Mappings (continued)



12) Add the following expression to the two attributes of the TOTAL constant operator:

Attribute	Expression
TOT	'Product Total'
SRC_ID	'99999'

Answer:

- On the canvas, click the TOTALS constant operator, and select the TOT attribute. In the Property Inspector window on the left, click the field next to Expression and enter 'Products Total'.

- On the canvas, with the TOTALS constant operator selected, click the SRC_ID attribute. In the Property Inspector window, click the field next to Expression and enter '99999'.

Solutions for Practice 7-5: Design ETL Mappings (continued)

The screenshot shows the Oracle Warehouse Builder interface. On the left, the 'Attribute Properties' pane is open for the attribute 'TOTALS.OUTGRP1.SRC_ID'. The 'SRC_ID' section is selected, showing an 'Expression' field containing '99999' with a red border around it. Below it, the 'Data Type Information' section shows 'VARCHAR2' as the data type. On the right, the 'Mapping' pane shows a connection from the 'SRC_ID' attribute to the 'OUTGRP1' operator in the 'TOTALS' category.

13) Change the FILTER operator to CATS.

Answer:

1. Right-click the FILTER operator, and select Open Details.
2. In the Filter Editor, ensure that the Name tabbed page is displayed. In the Name field, change FILTER to CATS, and click OK.

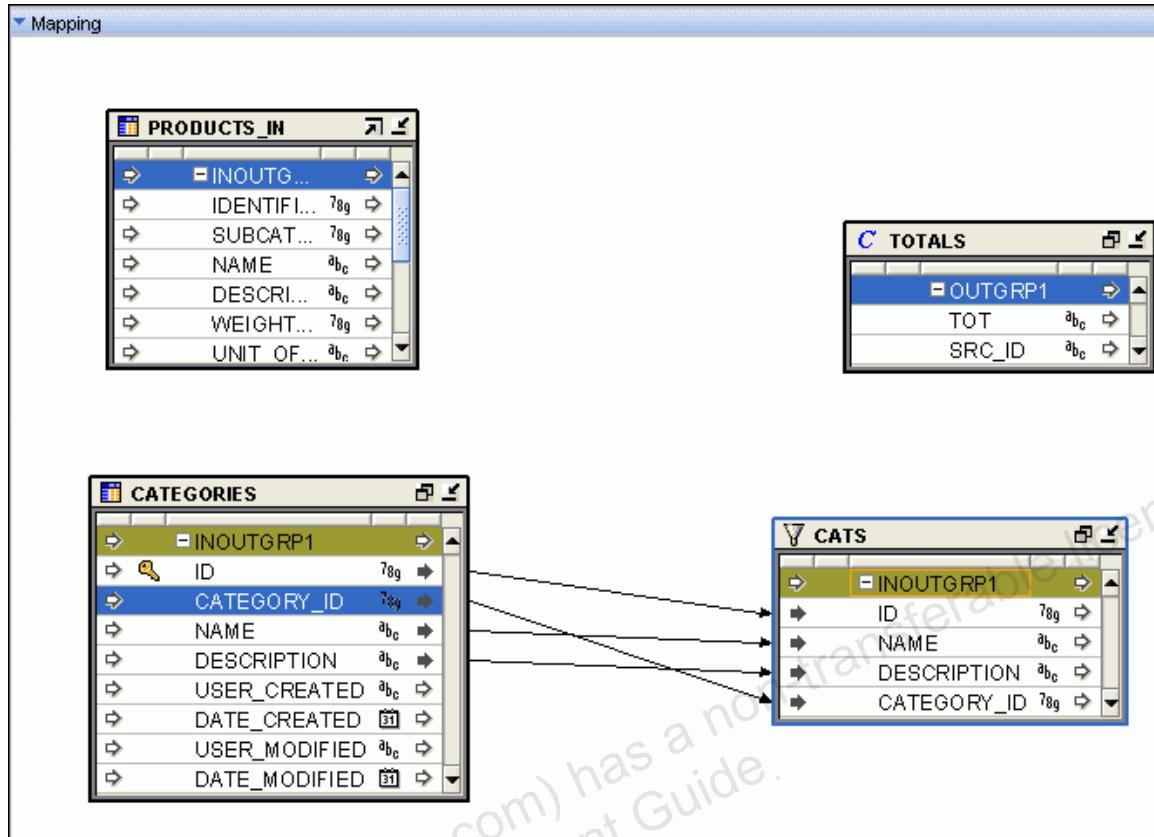
14) Add connection lines from attributes in the CATEGORIES.INOUTGRP1 operator to INOUTGRP1 of the CATS operator as:

CATEGORIES.INOUTGRP1	CATS
ID	INOUTGRP1
NAME	INOUTGRP1
DESCRIPTION	INOUTGRP1
CATEGORY_ID	INOUTGRP1

Answer:

1. Maximize the CATEGORIES operator so that all the attributes are clearly visible.
2. Draw connection lines as shown in the screenshot:

Solutions for Practice 7-5: Design ETL Mappings (continued)

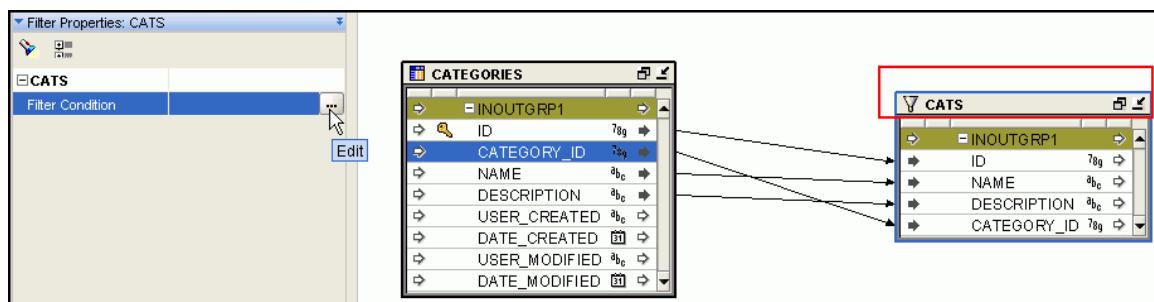


- 15) Specify the filter condition for the CATS filter operator as:

INOUTGRP1. CATEGORY_ID IS NULL

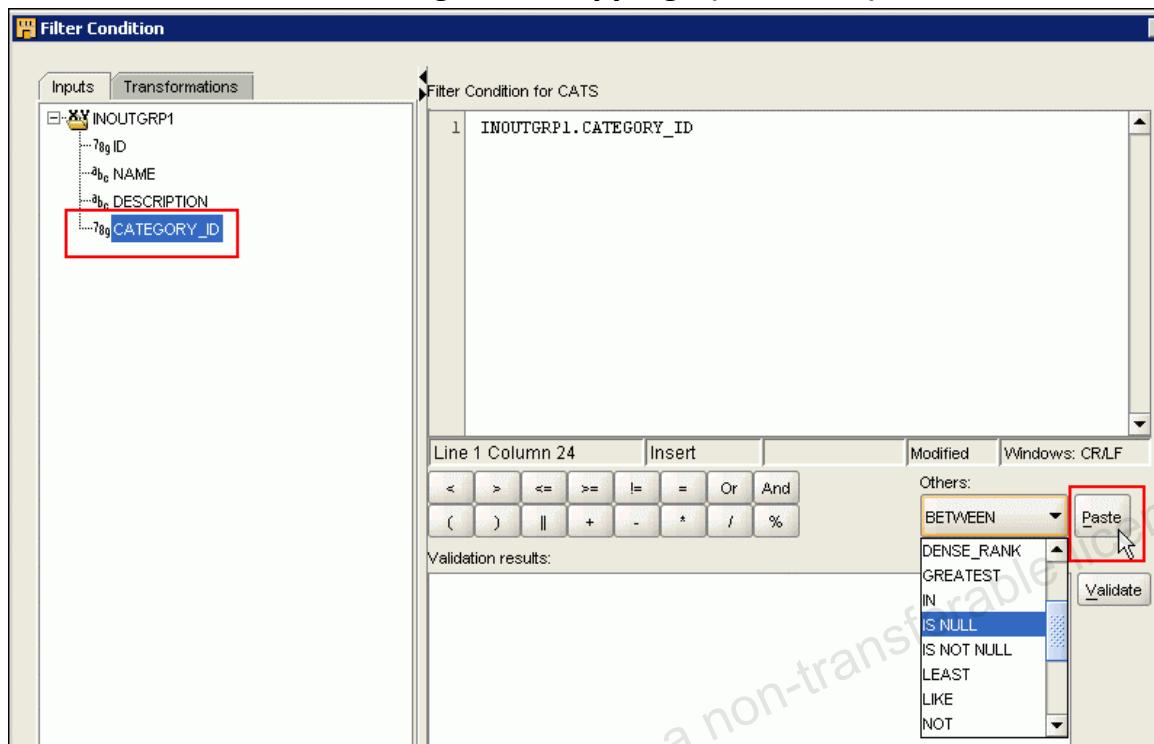
Answer:

- On the canvas, click the header area of the CATS filter operator. On the left, in the Filter Properties: CATS window, click the field next to Filter Condition. Click the three-dotted button (...). The Expression Builder for the filter condition is launched. OWB enables you to build the condition or enter the filter condition.

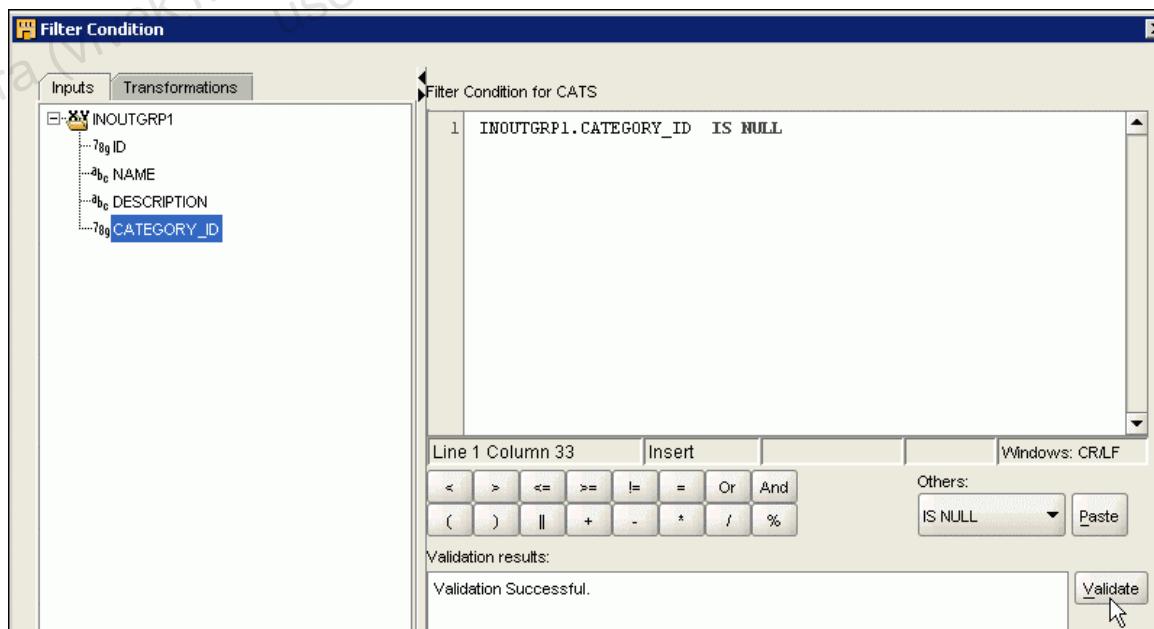


- In the Expression builder on the left, expand INOUTGRP1. Double-click CATEGORY_ID. You see that INOUTGRP1. CATEGORY_ID is pasted in the Filter condition field on the right.

Solutions for Practice 7-5: Design ETL Mappings (continued)



3. In the lower-right area, from the Others drop-down list, select IS NULL, and click Paste. You see the expression INOUTGRP1. CATEGORY_ID IS NULL in the Filter Condition for CATS field.
4. Click Validate. You see the result in the “Validation results” field. If the validation is successful, click OK.



- 16) Change the FILTER operator to SUBCATS.

Answer:

Solutions for Practice 7-5: Design ETL Mappings (continued)

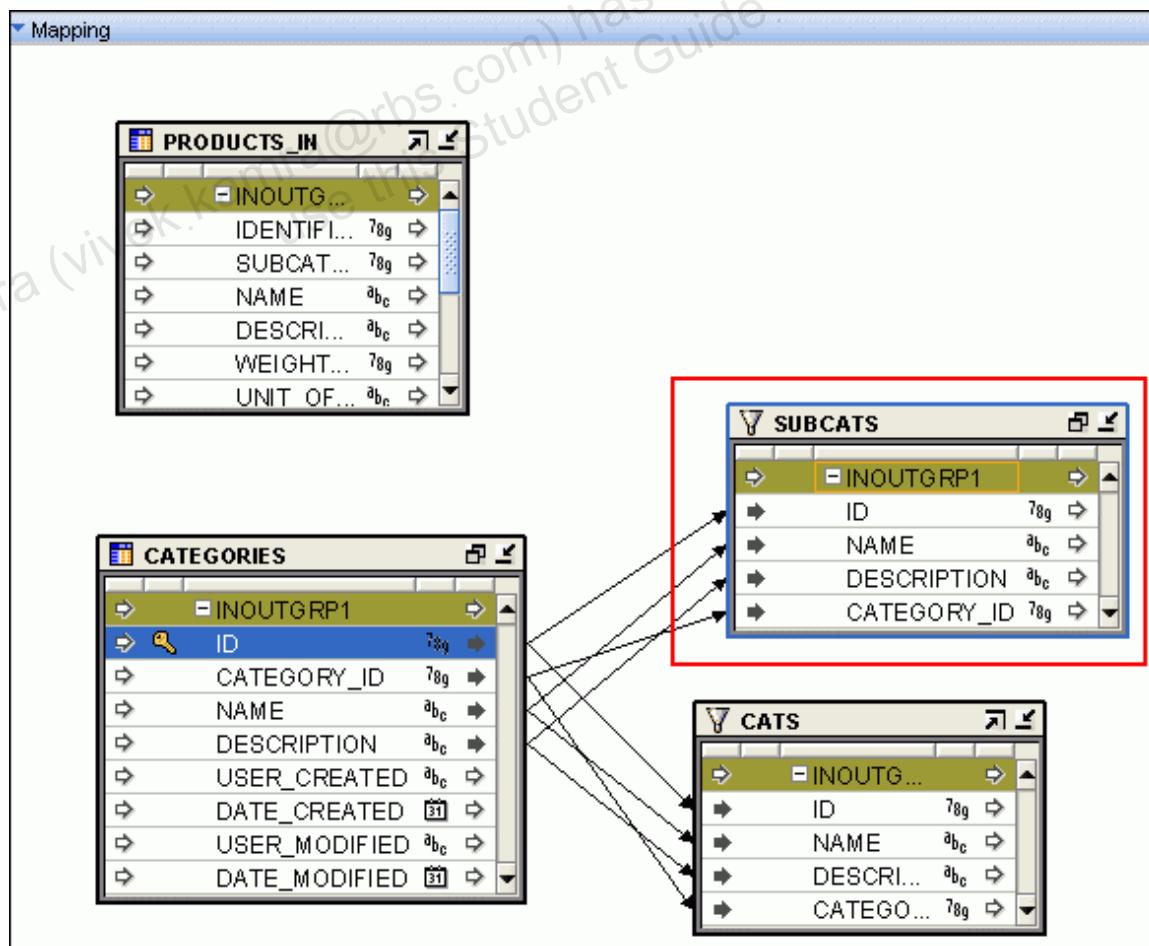
Right-click the FILTER operator, and select Open Details. In the Filter Editor, ensure that the Name tabbed page is displayed. In the Name field, change FILTER to SUBCATS, and click OK.

- 17) Add connection lines from attributes in the CATEGORIES.INOUTGRP1 operator to INOUTGRP1 of the SUBCATS operator as:

CATEGORIES.INOUTGRP1	SUBCATS
ID	INOUTGRP1
NAME	INOUTGRP1
DESCRIPTION	INOUTGRP1
CATEGORY_ID	INOUTGRP1

Answer:

Draw connection lines from attributes in the CATEGORIES.INOUTGRP1 operator to INOUTGRP1 of the SUBCATS operator as shown in the table. You can also look at the screenshot here:



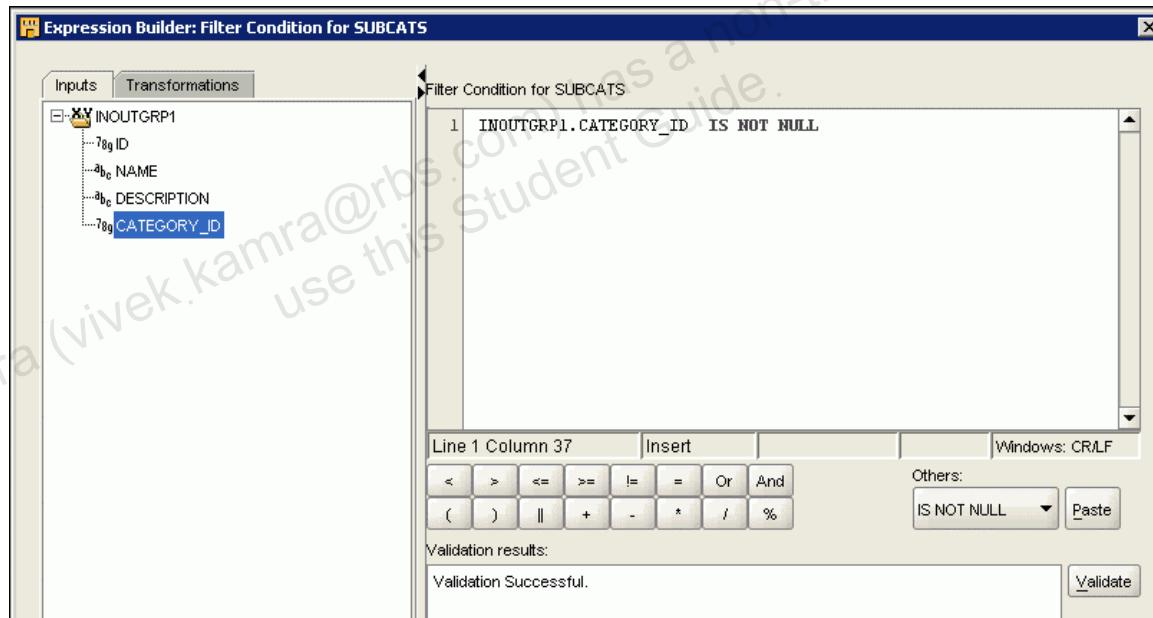
Solutions for Practice 7-5: Design ETL Mappings (continued)

- 18) Specify the filter condition for the SUBCATS filter operator as:

INOUTGRP1. CATEGORY_ID IS NOT NULL

Answer:

1. On the canvas, click the header area of the SUBCATS filter operator. On the left, in the Filter Properties: SUBCATS window, click the field next to Filter Condition. Click the three-dotted button (...). The Expression Builder for the filter condition is launched. OWB enables you to build the condition or enter the filter condition.
2. In the Expression Builder on the left, expand INOUTGRP1. Double-click CATEGORY_ID. You see that INOUTGRP1. CATEGORY_ID is pasted in the Filter condition field on the right.
3. In the lower-right section, from the Others drop-down list, select IS NOT NULL, and click Paste. You see the expression INOUTGRP1. CATEGORY_ID IS NOT NULL in the Filter Condition for SUBCATS field.
4. Click Validate. You see the result in the “Validation results” field. If the validation is successful. Click OK.



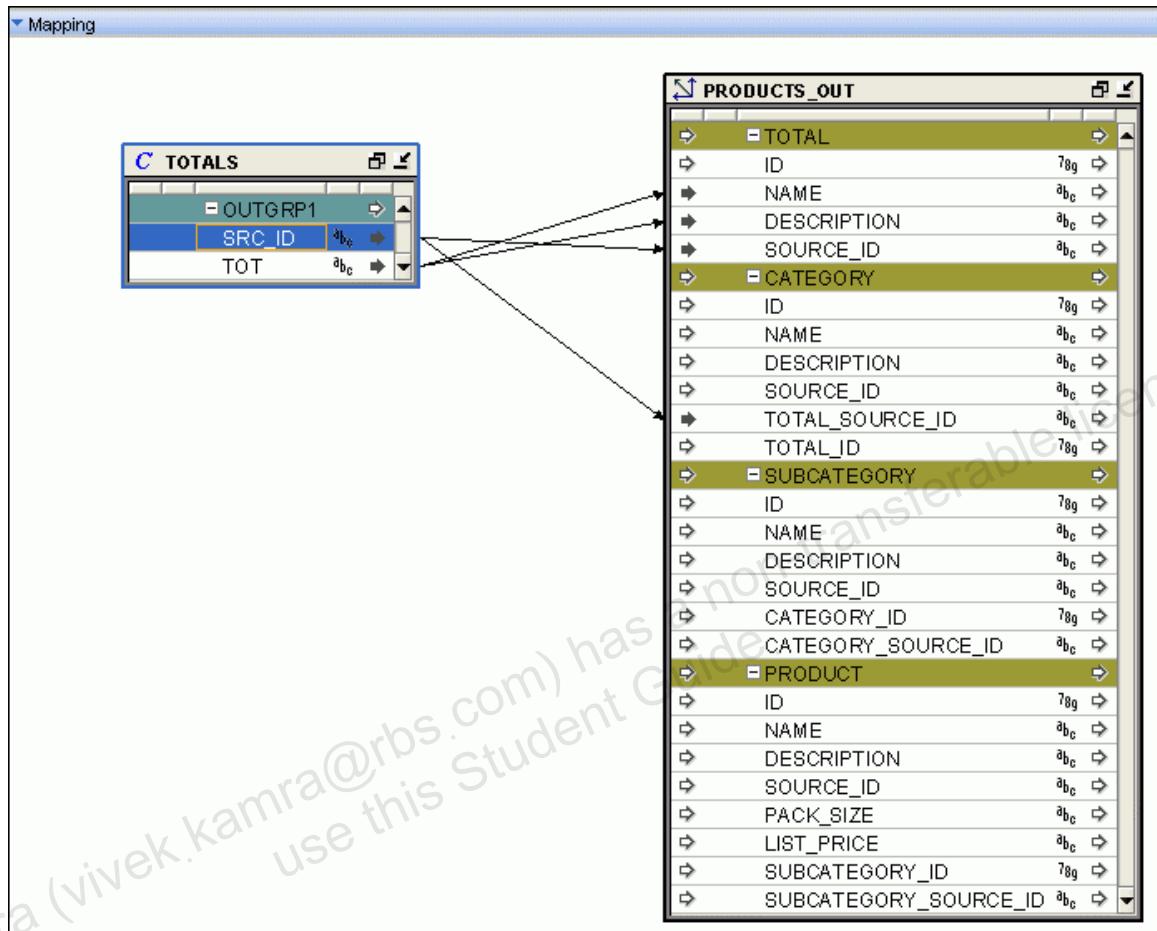
- 19) Draw connection lines from attributes in OUTGRP1 of the TOTALS constant operator to attributes in groups in the PRODUCTS_OUT dimension operator as:

TOTALS.OUTGRP1	PRODUCTS_OUT
TOT	TOTAL.NAME
TOT	TOTAL.DESCRIPTION
SRC_ID	TOTAL.SOURCE_ID
SRC_ID	CATEGORY.TOTAL_SOURCE_ID

Answer:

Solutions for Practice 7-5: Design ETL Mappings (continued)

On the canvas, draw connection lines from attributes in OUTGRP1 of the TOTALS constant operator to attributes in groups in the PRODUCTS_OUT dimension as shown in the following screenshot:



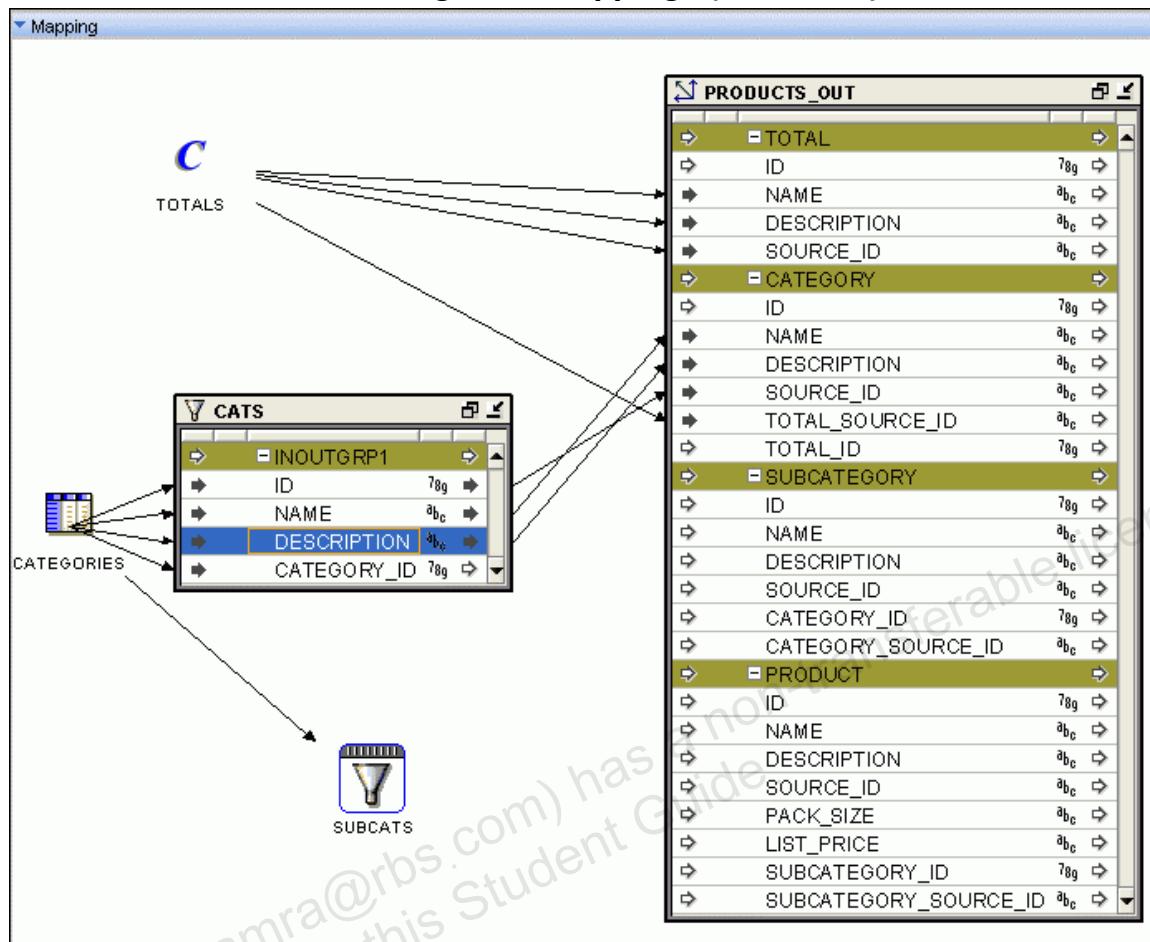
- 20) Draw connection lines from attributes in INOUTGRP1 of the CATS filter operator to attributes in the CATEGORY group of the PRODUCTS_OUT dimension operator.

CATS.INOUTGRP1	PRODUCTS_OUT
ID	CATEGORY.SOURCE_ID
NAME	CATEGORY.NAME
DESCRIPTION	CATEGORY.DESCRIPTION

Answer:

1. On the canvas, draw connection lines from attributes in INOUTGRP1 of the CATS filter operator to attributes in the CATEGORY group of the PRODUCTS_OUT dimension operator as shown in the following screenshot:

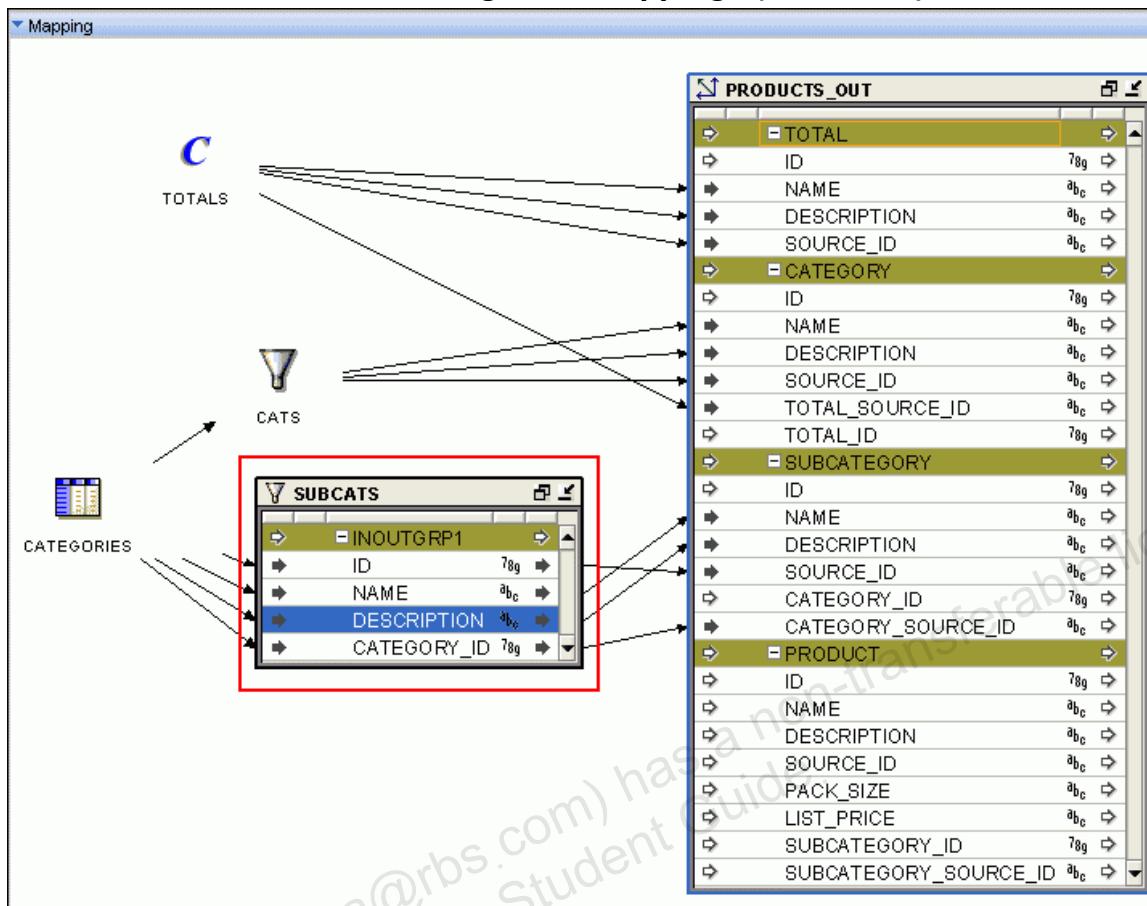
Solutions for Practice 7-5: Design ETL Mappings (continued)



- 21) Draw connection lines from attributes in INOUTGRP1 of the SUBCATS filter operator to attributes in the SUBCATEGORY group of the PRODUCTS_OUT dimension operator.

SUBCATS.INOUTGRP1	PRODUCTS_OUT
CATEGORY_ID	SUBCATEGORY.CATEGORY_SOURCE_ID
ID	SUBCATEGORY.SOURCE_ID
NAME	SUBCATEGORY.NAME
DESCRIPTION	SUBCATEGORY.DESCRIPTION

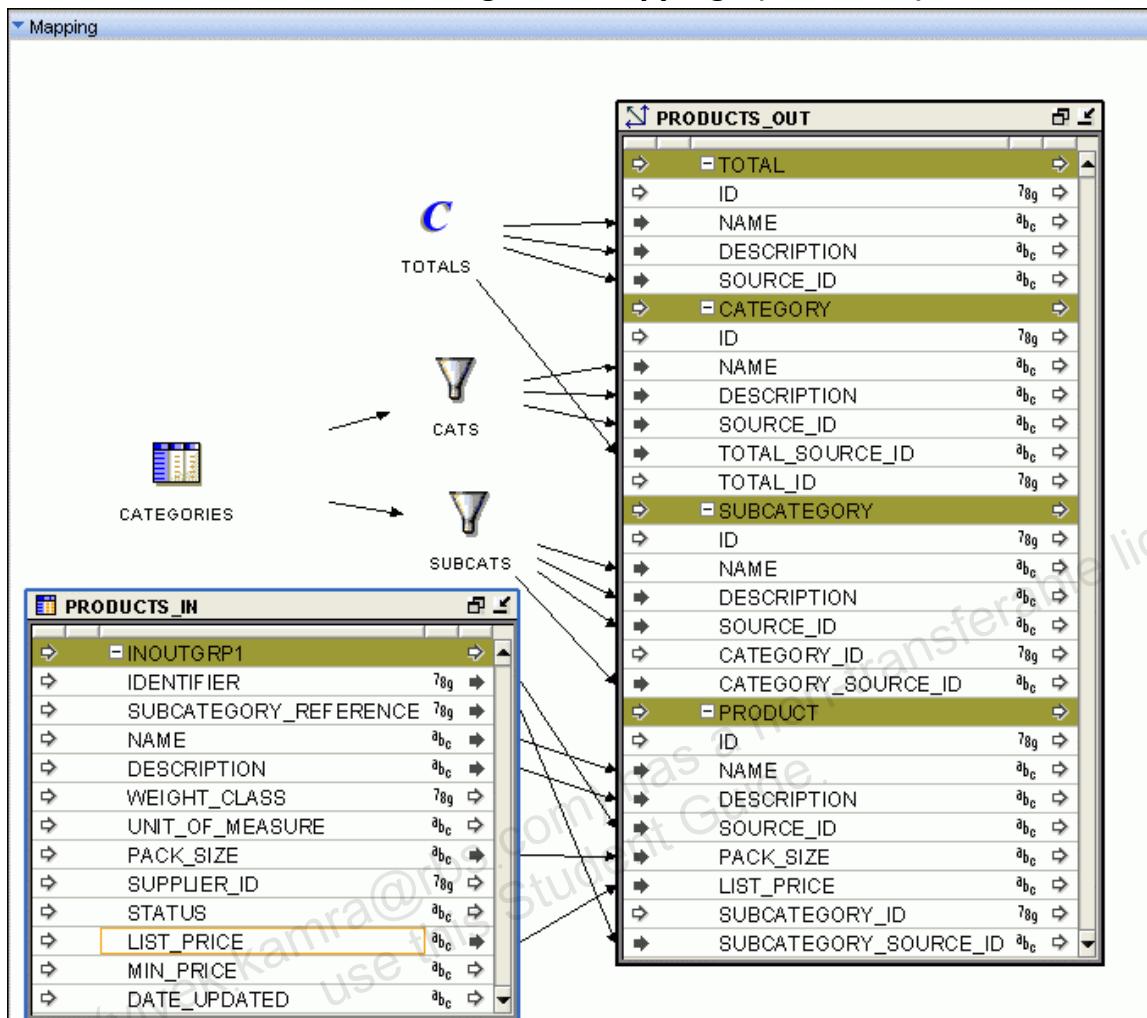
Solutions for Practice 7-5: Design ETL Mappings (continued)



- 22) Draw connection lines from attributes in INOUTGRP1 of the PRODUCTS_IN table operator to attributes in the PRODUCT group of the PRODUCTS_OUT dimension operator.

PRODUCTS_IN.OUTGRP1	PRODUCTS_OUT
SUBCATEGORY_REFERENCE	PRODUCT.SUBCATEGORY_SOURCE_ID
IDENTIFIER	PRODUCT.SOURCE_ID
NAME	PRODUCT.NAME
DESCRIPTION	PRODUCT.DESCRIPTION
PACK_SIZE	PRODUCT.PACK_SIZE
LIST_PRICE	PRODUCT.LIST_PRICE

Solutions for Practice 7-5: Design ETL Mappings (continued)



Validate the Mapping

23) Validate the LOAD_PRODUCTS mapping.

Answer:

In the Mapping Editor, select Validate from the Mapping menu. The Generation Results panel appears. Ignore the validation warnings.

Generate the Mapping

24) Generate the LOAD_PRODUCTS mapping.

Answer:

Select Generate from the Mapping menu. This will generate the code required to implement the design. Ignore the warnings (7 warnings). Close the Generation Results window.

25) Save your work.

26) Close the Mapping Editor.

You completed creating the mapping to load the PRODUCTS dimension.

Solutions for Practice 7-5: Design ETL Mappings (continued)

Complete the Partially Created CLASS_PROJECT

You completed designing the CHANNELS dimension, the PRODUCTS dimension, and the TIMES dimension. You then created the SALES cube and finally created the LOAD_PRODUCTS mapping to load the PRODUCTS dimension. The other objects were created for you at the initial setup. To save time during this class, the instructions to design the mapping that loads the cube is precreated for you. For your reference, the mapping to load the SALES cube is found in the appendix titled “MOLAP: Modeling and Loading.”

Run the tcl Scripts to Complete the Design of CLASS_PROJECT

- 27) In the OMB Plus panel in the Design Center, execute the maps.tcl script to complete the design of this project.

Answer:

1. From the Windows Start menu, select Programs > Oracle > OWBHome10gR2 > Warehouse Builder > OMB Plus. The OMB Plus command-line window appears.
2. At the OMB+> prompt, enter the following:

```
OMB+> cd e:/labs/tcl  
OMB+> source maps.tcl
```

The screenshot shows a command-line window titled "OMB*Plus". The user has entered the commands "cd e:/labs/tcl" and "source maps.tcl". The output of the script is displayed below, showing the creation and alteration of various mappings:

```
OMB+> cd e:/labs/tcl  
OMB+> source maps.tcl  
Context changed.  
Loading ROLAP Maps  
Map LOAD_CHANNELS created.  
Map LOAD_CHANNELS altered.  
Map LOAD_CHANNELS altered.  
Map LOAD_CHANNELS altered.  
Map LOAD_CHANNELS altered.  
Map LOAD_CHANNELS created  
Map LOAD_SALES created.  
Map LOAD_SALES altered.  
Map LOAD_SALES altered.  
Map LOAD_SALES created  
Commit complete.  
Commit complete.  
OMB+>
```

- 28) Save your work.

Now you have all the mappings needed to complete the design of the warehouse. You use the Generate option to generate the code to implement the design.

Generate the Remaining Mappings

You designed and generated the LOAD_PRODUCTS mappings.

- 29) Generate the following mappings.

Solutions for Practice 7-5: Design ETL Mappings (continued)

LOAD_PROMOTIONS

LOAD_CHANNELS

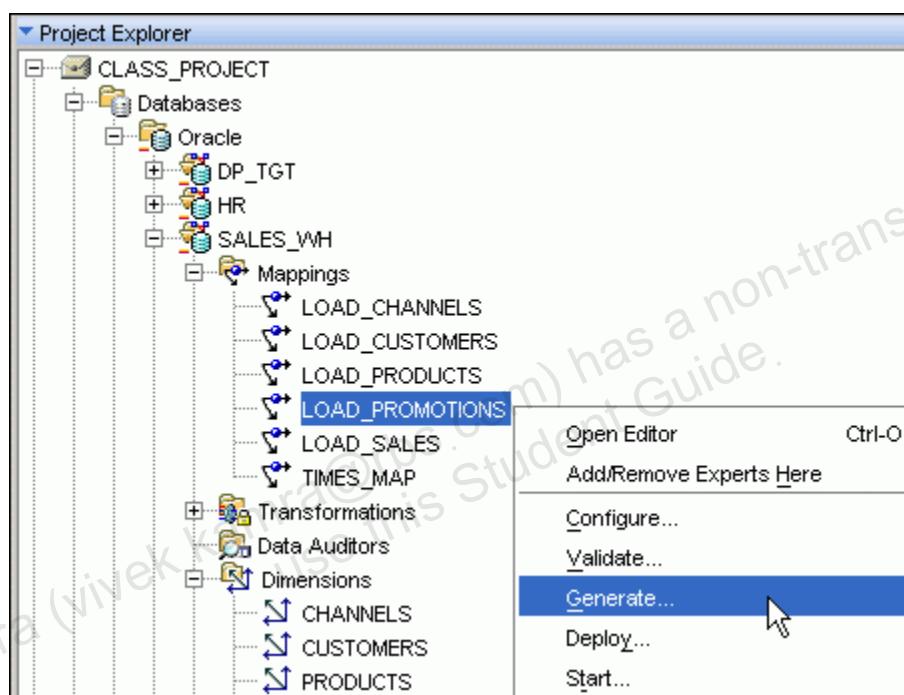
LOAD_CUSTOMERS

TIMES_MAP

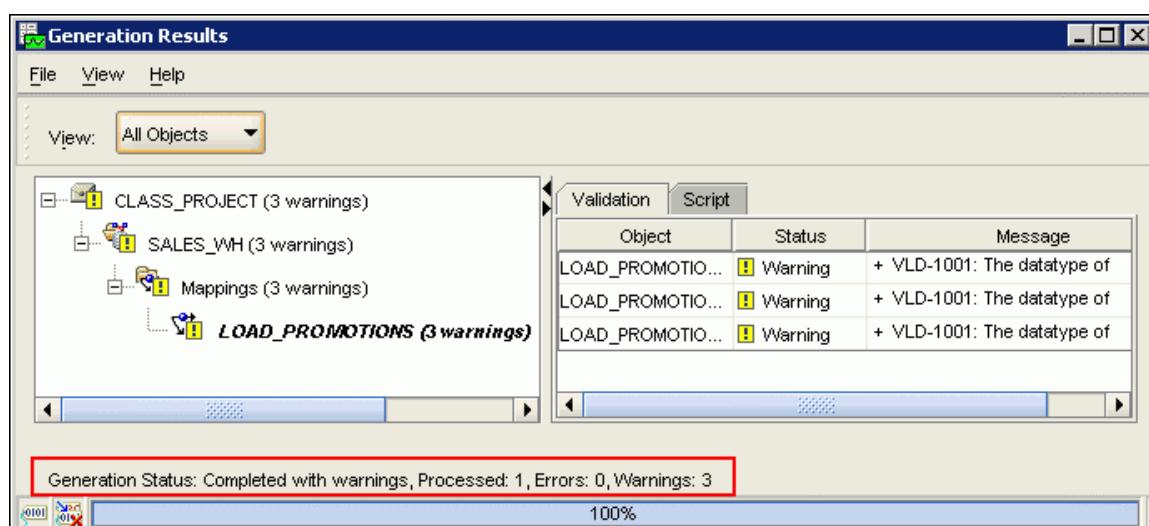
LOAD_SALES

Answer:

1. In the Design Center, expand SALES_WH > Mappings.
2. Right-click the LOAD_PROMOTIONS mapping, and select Generate.



3. The Generation Results window displays the results. Ignore the warnings. There should not be any errors. Close the Generation Results window.



Solutions for Practice 7-5: Design ETL Mappings (continued)

4. Repeat steps 2 and 3 with each of the remaining mappings.
 - a. LOAD_CHANNELS returns 1 warning.
 - b. LOAD_CUSTOMERS returns 9 warnings.
 - c. LOAD_SALES returns 5 warnings.
 - d. TIMES_MAP returns no warnings.

Practice Solutions for Lesson 8

In this practice, you learn how to create a Type 2 Slowly Changing Dimension.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 8-1: Handling Slowly Changing Dimensions

In this practice, you use the existing CHANNELS dimension and edit the SCD type from type 1 to type 2.

- 1) Open the CHANNELS dimension in the Data Object Editor.

Answer:

In the Design Center, expand SALES_WH > Dimension. Double-click CHANNELS. The Data Object Editor is launched.

- 2) Add the following two attributes:

Name	Data type
Effective_Date	DATE
Expiration_Date	DATE

Answer:

1. On the Dimension Details page, click the **Attributes** tab. Add the following two attributes as shown in the screenshot.

The screenshot shows the 'Dimension Details: SALES_WH.CHANNELS' window. The 'Attributes' tab is selected. A message at the top says 'Choose the sequence that will populate the Dimension and Surrogate Keys:' followed by 'CHANNELS_SEQ'. Below is a table of attributes:

	Name	Description	Identifier	Data Type	Length
1	ID		Surrogate	NUMBER	
2	NAME			VARCHAR2	60
3	SOURCE_ID		Business	VARCHAR2	40
4	EFFECTIVE_DATE			DATE	
5	EXPIRATION_DATE			DATE	

- 3) On the Levels tabbed page, apply the two new attributes EFFECTIVE_DATE and EXPIRATION_DATE to the CHANNEL level.

Answer:

1. In the Dimension Details panel, click the Levels tab.
2. Select the CHANNEL level.
3. In the Level Attribute section, in the Dimension Attribute Name column, click EFFECTIVE_DATE. In the Applicable column, select the check box such that EFFECTIVE_DATE is applied to the CHANNEL level. Similarly, select EXPIRATION_DATE from the Level Attribute Name column and in the Applicable column, select the check box. Now the two attributes

Solutions for Practice 8-1: Handling Slowly Changing Dimensions (continued)

(EFFECTIVE_DATE and EXPIRATION_DATE) are applied to the CHANNEL level.

Level	Description
1	TOTAL
2	CLASS
3	CHANNEL

Dimension Attribute Name	Applica...	Level Attribute Name	De
ID	<input checked="" type="checkbox"/>	ID	
NAME	<input checked="" type="checkbox"/>	NAME	
SOURCE_ID	<input checked="" type="checkbox"/>	SOURCE_ID	
EFFECTIVE_DATE	<input checked="" type="checkbox"/>	EFFECTIVE_DATE	
EXPIRATION_DATE	<input checked="" type="checkbox"/>	EXPIRATION_DATE	

- 4) On the SCD tabbed page, select “Type 2: Store the complete change history.”
Note: The Settings button is enabled.

Answer:

In the Dimension Details panel, click the SCD tab. Select “Type 2: Store the complete change history.” The Settings button is now enabled.

Slowly changing dimensions determine how you store historical changes to your dimension values. If you want to store the complete history of changes, choose Type 2.

Specify slowly changing type for the dimension:

Type 1: Do not keep history
 Type 2: Store the complete change history
 Type 3: Store only the previous value

Settings...

- 5) Click the **Settings** button. In the “Type 2 slowly changing dimension” window, select the attributes that trigger history, and set the values for the two data attributes as:

**Solutions for Practice 8-1: Handling Slowly Changing Dimensions
(continued)**

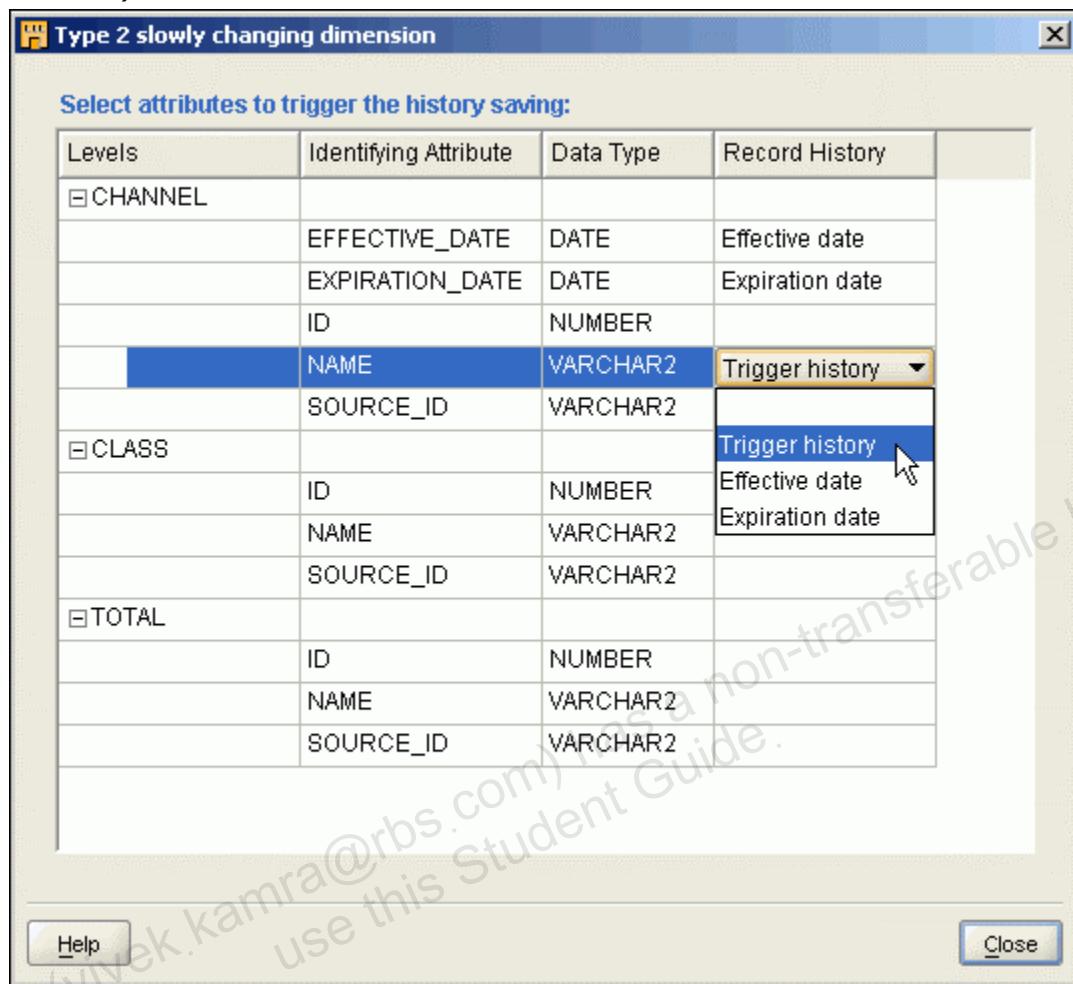
Levels	Identifying Attribute	Record History
CHANNEL	EFFECTIVE_DATE	Effective date
	EXPIRATION_DATE	Expiration date
	NAME	Trigger History

Answer:

1. Click Settings. The “Type two slowly changing policy” window appears.
2. In the “Type two slowly changing policy” window, the CHANNEL level has the additional two date attributes (EFFECTIVE_DATE and EXPIRATION_DATE). In the Levels column, click CHANNEL. In the Identifying Attribute column, click EFFECTIVE_DATE. In the Record History column, select “Effective date” from the drop-down list. Click EXPIRATION_DATE, and in the Record History column, from the drop-down list, select “Expiration date.”
3. Ensure that the Channel level is selected. Click NAME in the Record History column. Select “Trigger history” from the drop-down list.

Note: In the CHANNELS dimension, SOURCE_ID is the business key. The attribute that acts as the triggering history must be any attribute other than the surrogate ID and the business key.

Solutions for Practice 8-1: Handling Slowly Changing Dimensions (continued)



- Click Close to close the “Type two slowly changing policy” window.

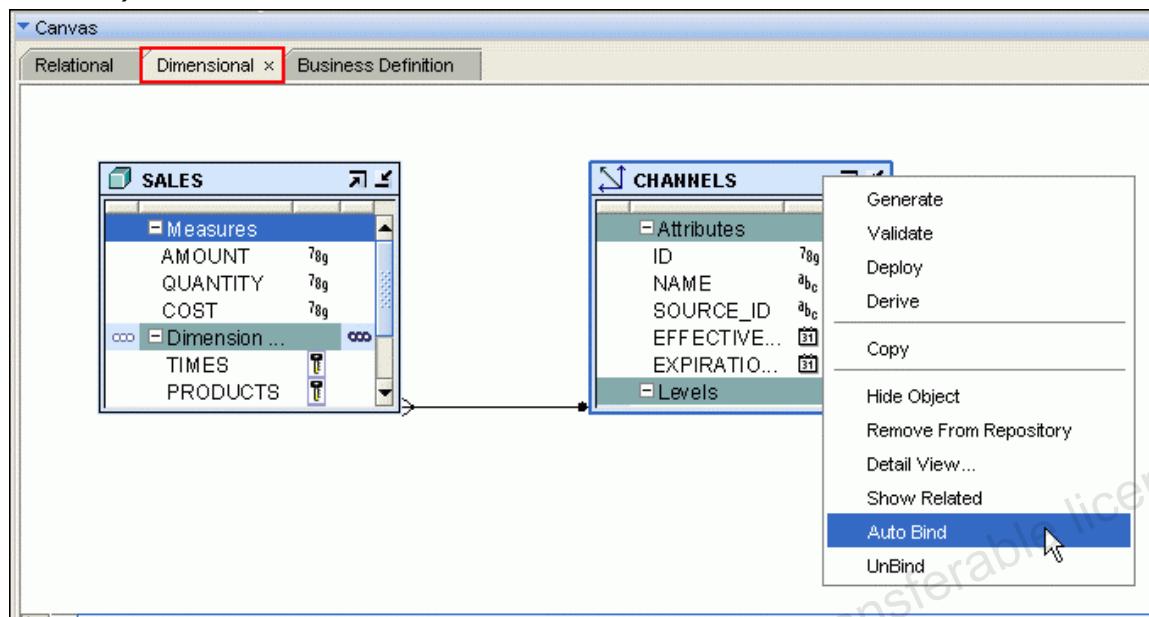
Binding the Dimension Channels to the Database Table

- Propagate the changes made to the dimension object to the database table.

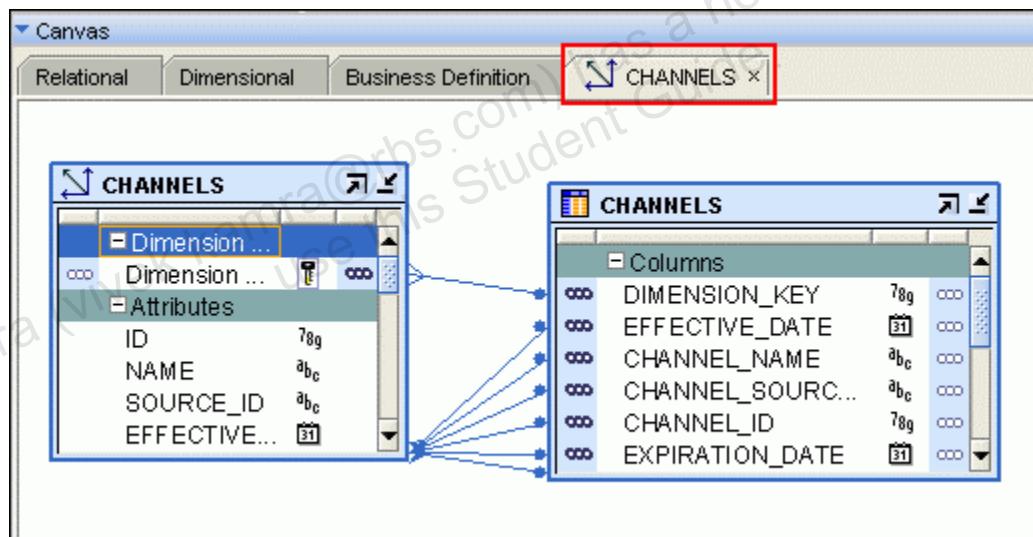
Answer:

- On the Canvas panel, in the Data Object Editor, ensure that the Dimensional tabbed page is displayed. Right-click the header area of the CHANNELS dimension, and select Auto Bind.

Solutions for Practice 8-1: Handling Slowly Changing Dimensions (continued)



2. You see that the two attributes are now reflected in the database table as well.



- 7) Save your work, and close the Data Object Editor.

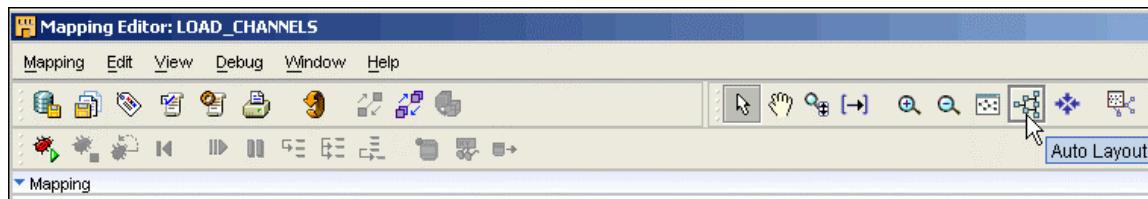
Synchronize the CHANNELS Repository Object with the CHANNELS Operator

- 8) In the Mapping Editor, the CHANNELS dimension operator needs to be updated with the new CHANNELS repository object.

Answer:

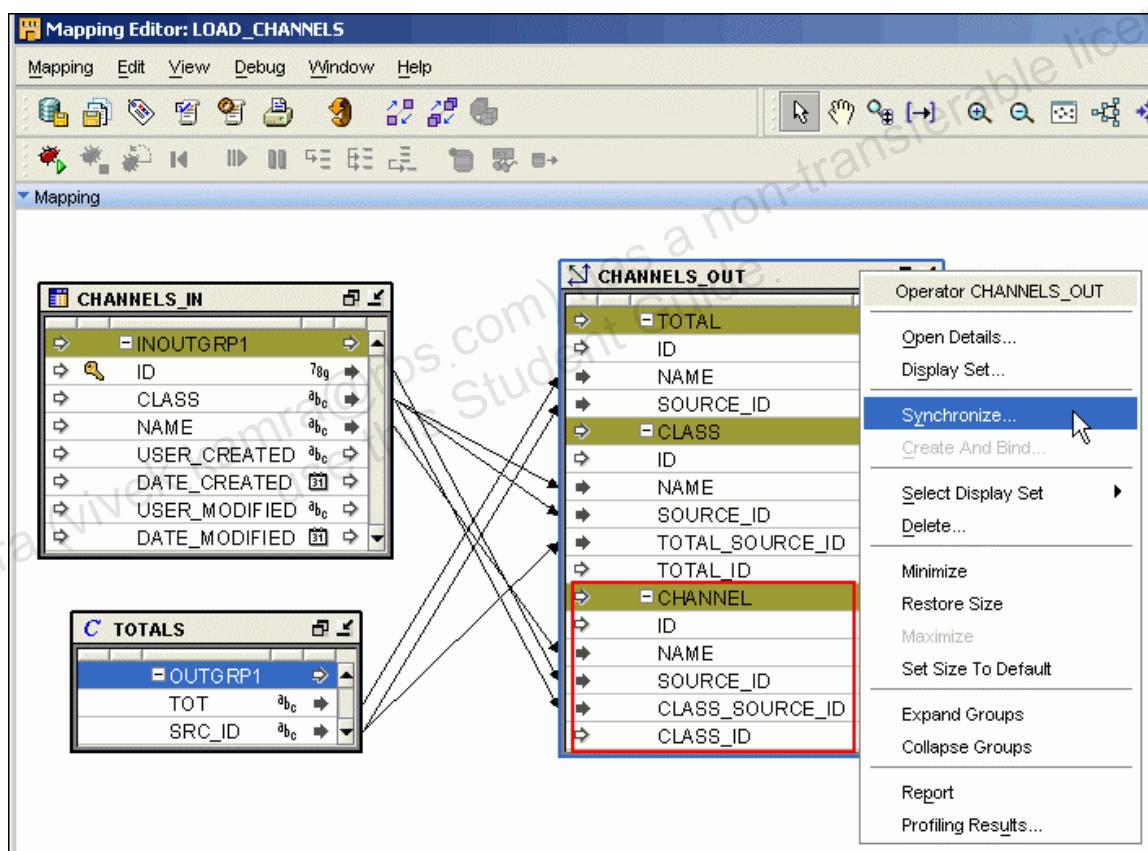
1. In the Design Center, expand SALES_WH > Mappings. Double-click LOAD_CHANNELS. The Mapping Editor is launched.
2. In the Mapping Editor if you are not able to see all the objects in the window, click the Auto Layout icon on the toolbar.

Solutions for Practice 8-1: Handling Slowly Changing Dimensions (continued)



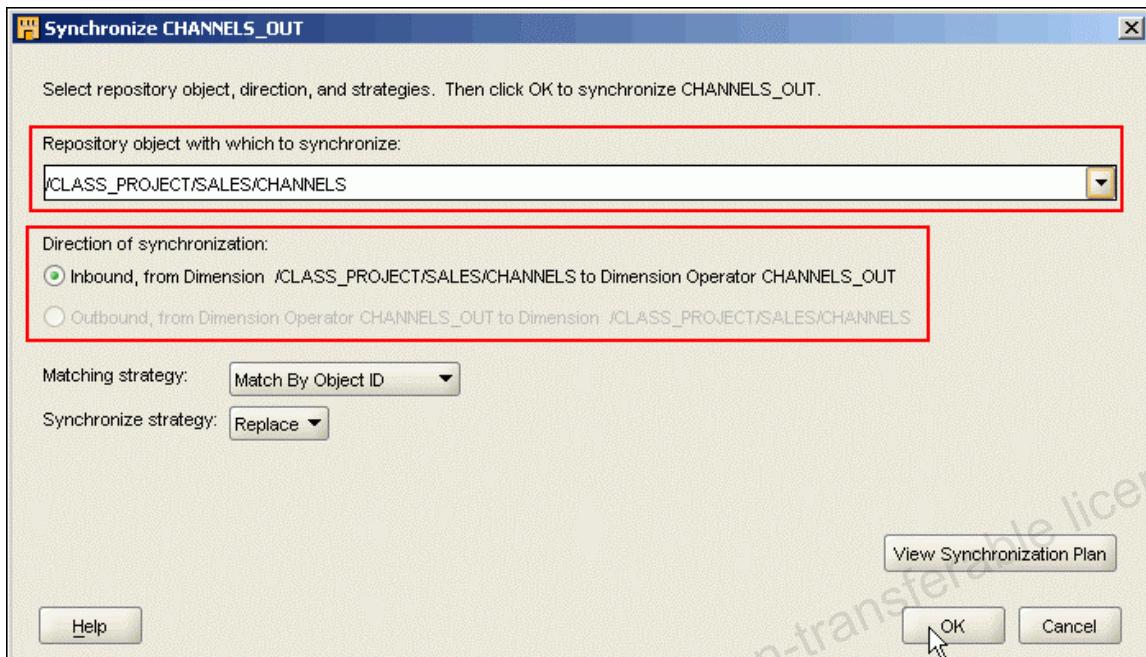
- Right-click the header area of the CHANNELS_OUT dimension operator, and select Synchronize. It will synchronize this dimension with the underlying repository object.

Note: Observe this screenshot carefully. In the CHANNELS_OUT dimension operator, under the CHANNEL level, you do not see the two date attributes. These will be reflected after you complete the synchronize process.

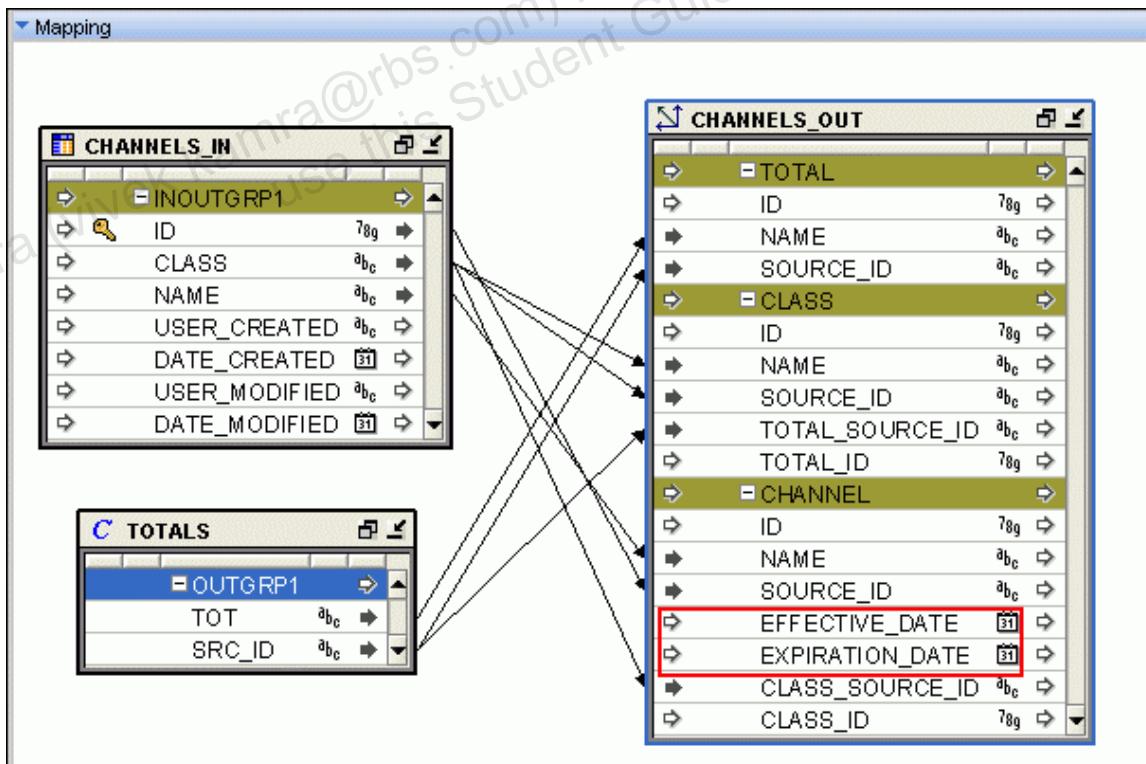


- In the Synchronize dialog box, accept the default options, and click OK. You are going to synchronize from the repository object to the dimension operator.

Solutions for Practice 8-1: Handling Slowly Changing Dimensions (continued)



5. Note the changes after the synchronize process is complete. The dimension operator in the mapping editor reflects the changes.



- 9) Save your work.
10) Close the Mapping Editor

Practice Solutions for Lesson 9

On the basis of the business intelligence needs, you can specify how you want to deploy your design (for example, deploy to Relational Databases, Oracle Workflow, or Oracle Discoverer). In the previous lessons, you specified to deploy to an Oracle location (SALES_WH_LOCATION).

For this class, when working with process flows, you use Oracle Workflow location. Before you start designing the process flow, you create an Oracle Workflow location.

You define an Oracle Workflow location to specify where you want to deploy your business definitions. The Oracle Workflow location will point to a workflow engine (OWF_MGR) that runs in the target database. Warehouse Builder process flows comply with the XML Process Definition Language (XPDL).

Note: You must run the Workflow Configuration Assistant to load Oracle Workflow into your database.

For this classroom setup, Oracle Workflow has been loaded into your current working database. OWF_MGR is the username of your Oracle Workflow database account. The default Workflow account for a fresh installation is OWF_MGR.

The next step that you have to perform is to create an Oracle Workflow location.

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow

Create an Oracle Workflow Location

- 1) Create an Oracle Workflow location named OWF_MGR_LOCATION with the following details:

Password: **owf_mgr**

Host: **localhost**

Port: **1521**

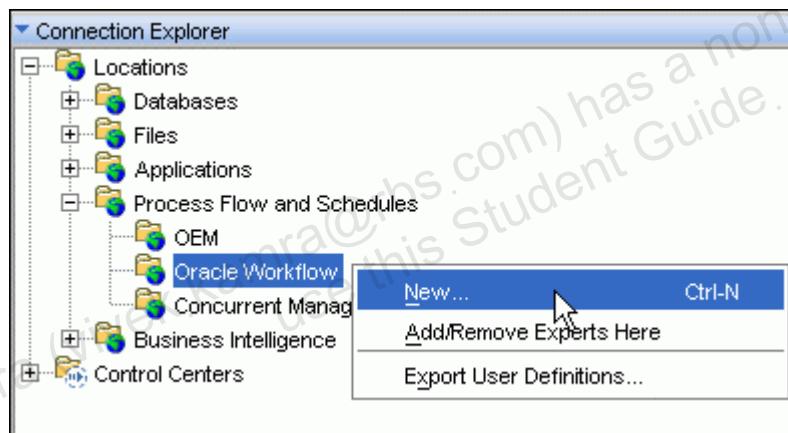
Service: **orcl**

Schema: **owf_mgr**

Version: **2.6.4**

Answer:

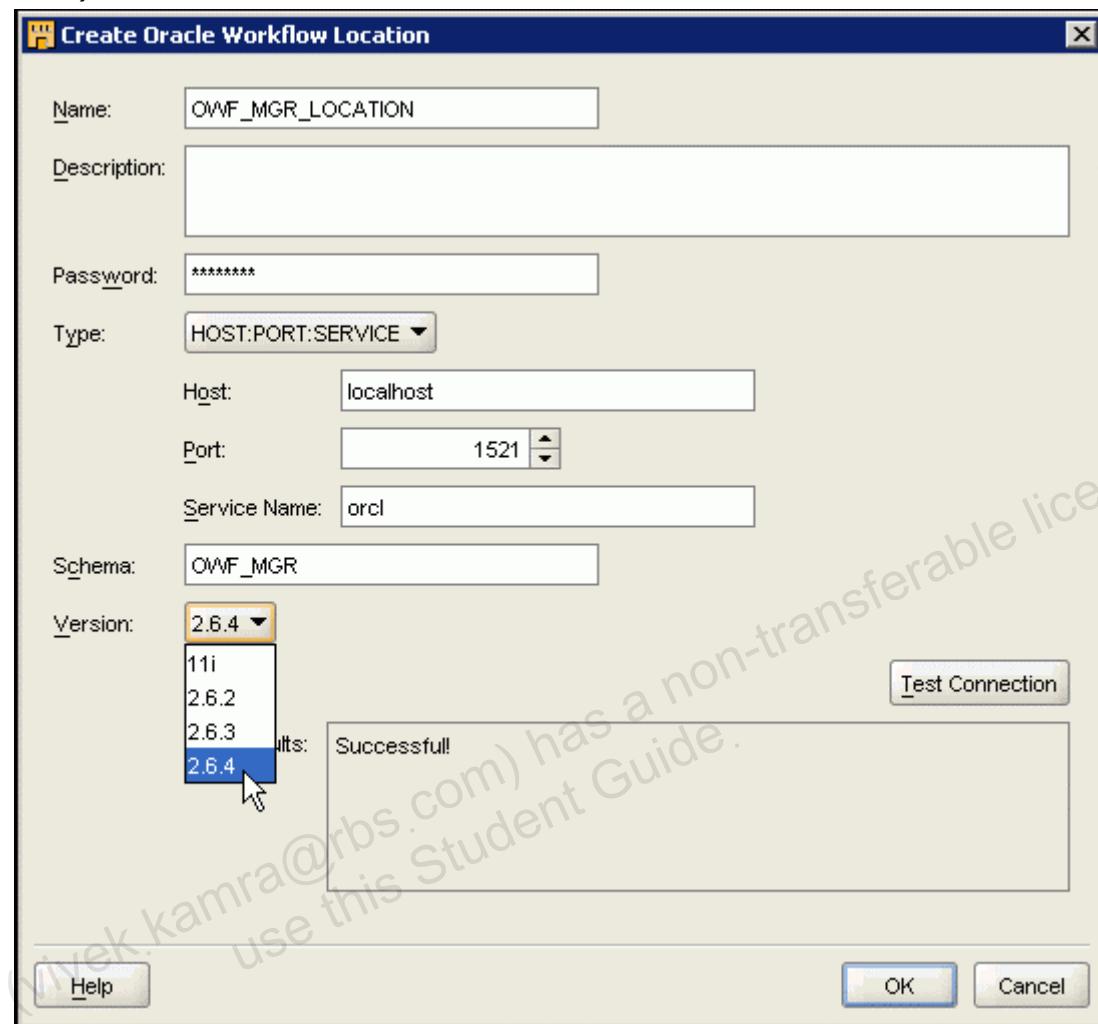
1. In the Design Center, in the Connection Explorer, expand **Locations > Process Flows and Schedules**. Right-click **Oracle Workflow**, and select **New**.



Note: The Create Oracle Workflow Location page appears. On this page, you specify the connection details to the Oracle Workflow schema. This course uses **owf_mgr** as schema name and **owf_mgr** as password.

2. Create the Oracle Workflow location with the details as shown in the screenshot. The password is **owf_mgr**. Click **Test Connection**. If the connection is successful, click **OK**.

**Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow
(continued)**

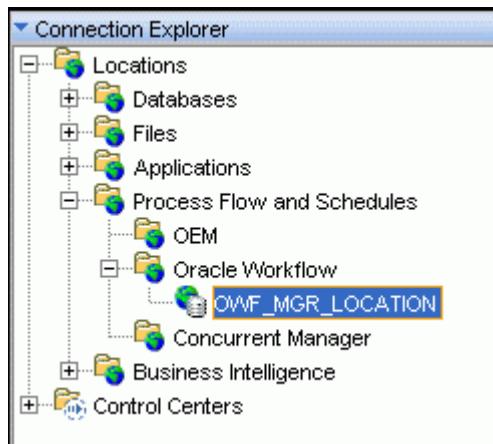


- 2) Verify whether this newly created OWF_MGR_LOCATION exists.

Answer:

6. In the Design Center, in the Connection Explorer panel, expand **Locations > Process Flows and Schedules**. You see OWF_MGR_LOCATION, the location that you just created.

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)



You have completed creating the Oracle Workflow location. You now design the process flow.

Grant privileges to the owf_mgr schema

- 3) Grant the OWB_O REP OWNER privileges to owf_mgr.

Answer:

1. Log in to SQL*Plus as sys/oracle as sysdba.
2. At the SQL prompt, enter: @e:\labs\sql\grant_role.sql

Design a Process Flow

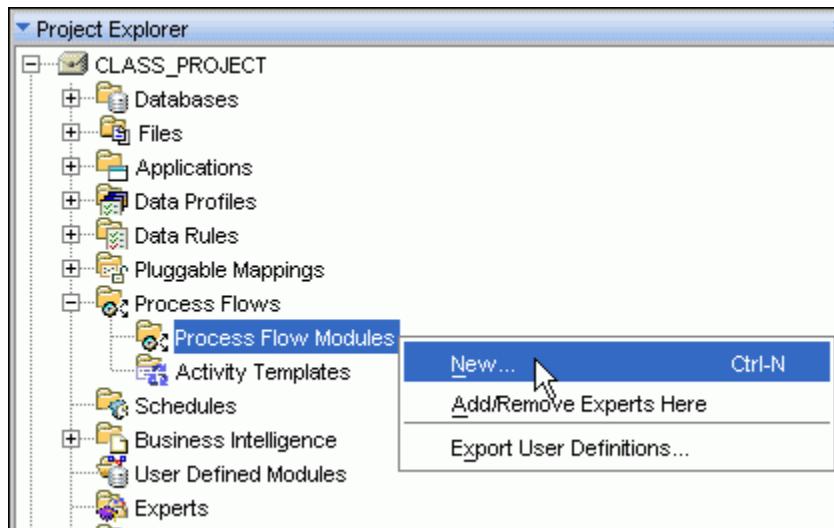
- 4) Create a process flow with the following details:

- a) Name: **PF_SALES_WH**.
- b) Description: **Process Flow module for SALES_WH**
- c) Location: **OWF_MGR_LOCATION**
- d) Process Flow Package: **PK_SALES**
- e) Process Flow: **LOAD_SALES_WH**

Answer:

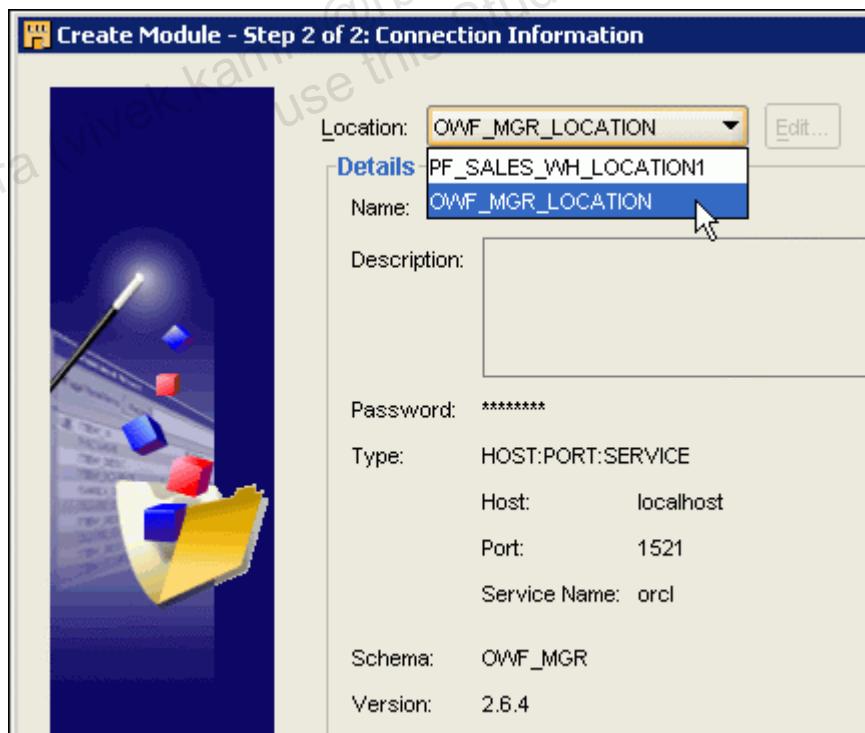
1. In the Design Center, in the Project Explorer, expand CLASS_PROJECT > Process Flows. Right-click Process Flow Modules, and select New. The Create Module Wizard is launched. Click Next on the Welcome page.

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)



2. On the Name and Description page, enter the following information:
 - a. Name: **PF_SALES_WH**
 - b. Description: **Process Flow module for SALES_WH**

Click Next.
3. On the Connection Information page, select **OWF_MGR_LOCATION** from the Location drop-down list. Click **Next**.



4. On the Summary page, review the summary details, and click **Finish**. The Create Process Flow Package dialog box pops up.

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)

5. In the Process Flow Package dialog box, enter **PK_SALES** as the name of the process flow package, and click **OK**. The Create Process Flow dialog box pops up.
6. In the Create Process Flow dialog box, enter **LOAD_SALES_WH** as the name of the process flow, and click **OK**.

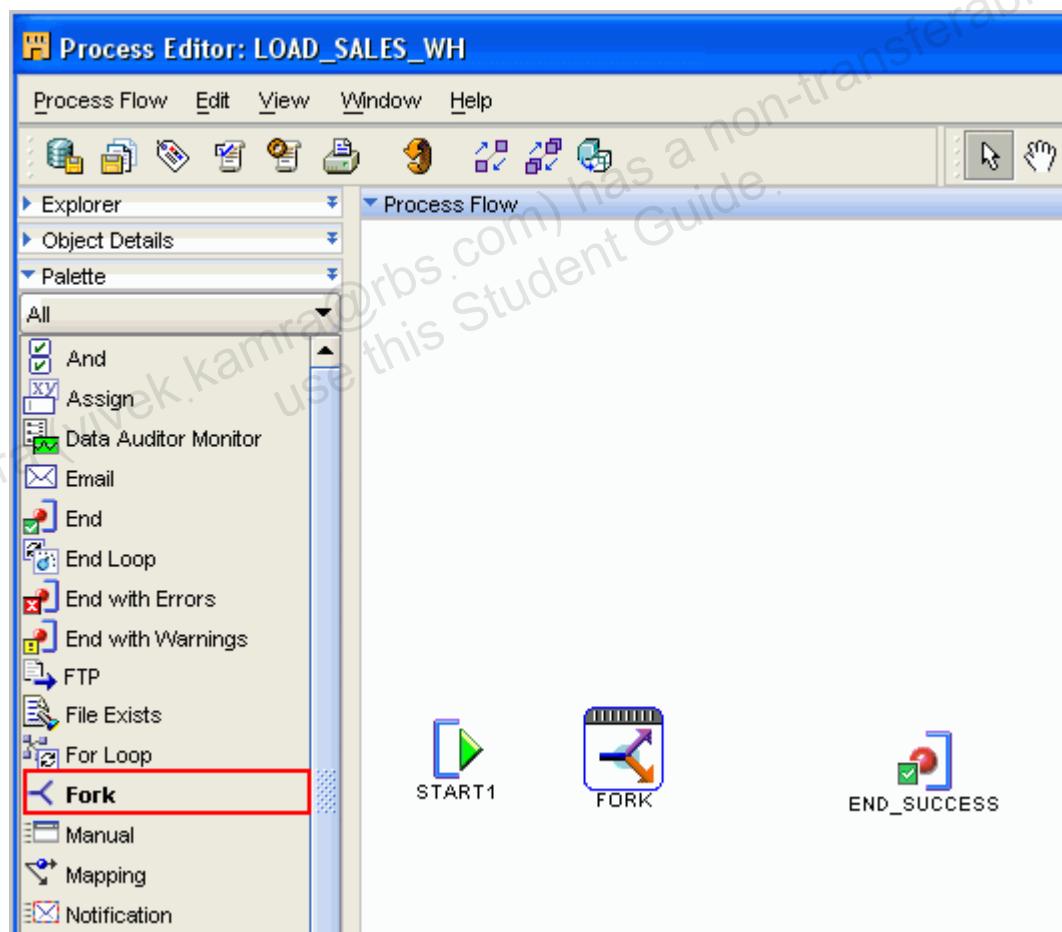
The process flow is created and the Process Editor is launched.

Add a Fork Activity

- 5) In the Process Editor, drag a Fork activity to the canvas. Place it between the START activity and the END activity.

Answer:

1. In the Process Editor, from the Palette, drag a **Fork** activity to the canvas. Place the FORK activity after the START activity and before the END activity.



Add the Mappings

- 6) Drag the following six mappings to the canvas:

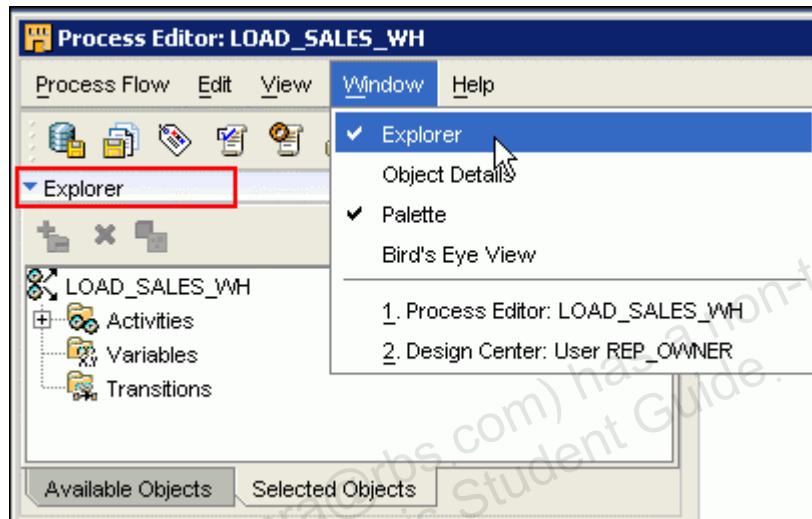
- LOAD_CHANNELS
- LOAD_PROMOTIONS

**Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow
(continued)**

- LOAD_CUSTOMERS
- LOAD_PRODUCTS
- TIMES_MAP
- LOAD_SALES

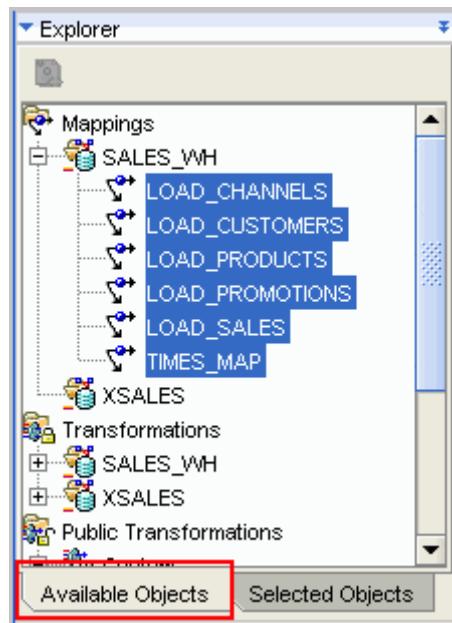
Answer:

1. In the Process Editor, from the Windows menu select Explorer (if not already selected). The Explorer panel should be visible.



2. In the Explorer panel, click the **Available Objects** tab, if it is not already selected. Expand **Mappings > SALES_WH**. Drag the following mappings to the canvas:
 - LOAD_CHANNELS
 - LOAD_PROMOTIONS
 - LOAD_CUSTOMERS
 - LOAD_PRODUCTS
 - TIMES_MAP
 - LOAD_SALES

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)



- 7) Arrange the activities on the process flow canvas such that the following mapping activities appear after the FORK activity and before the END activity:
- LOAD_CHANNELS
 - LOAD_PROMOTIONS
 - LOAD_CUSTOMERS
 - LOAD_PRODUCTS
 - TIMES_MAP

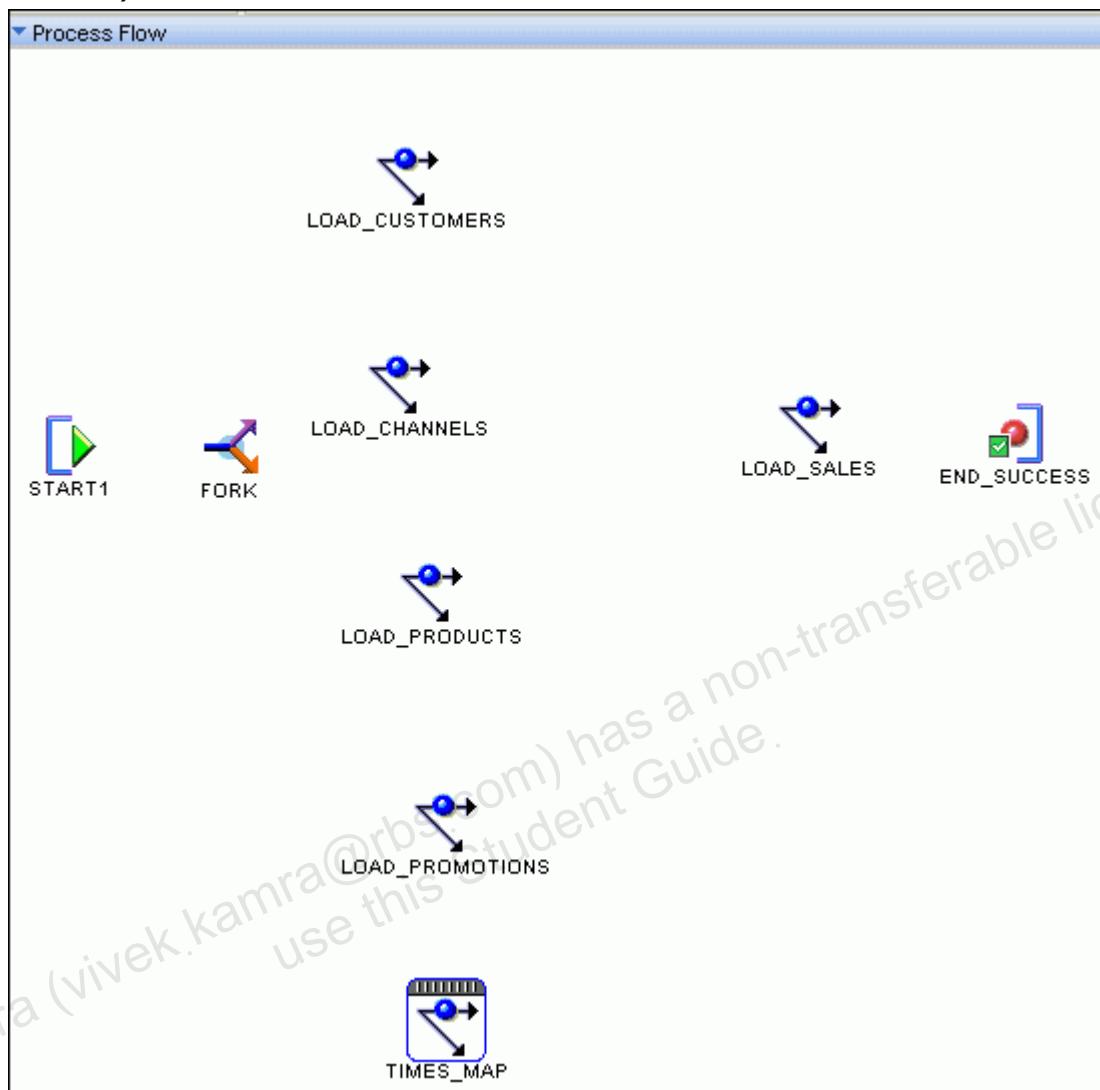
Answer:

1. To move an activity within the canvas, select the activity. Place the cursor in the header area of the activity. The cursor changes to a single-headed arrow. Click in the header area and keep the left mouse button clicked and drag the activity to the desired location.



2. Arrange the activities as follows:

**Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow
(continued)**



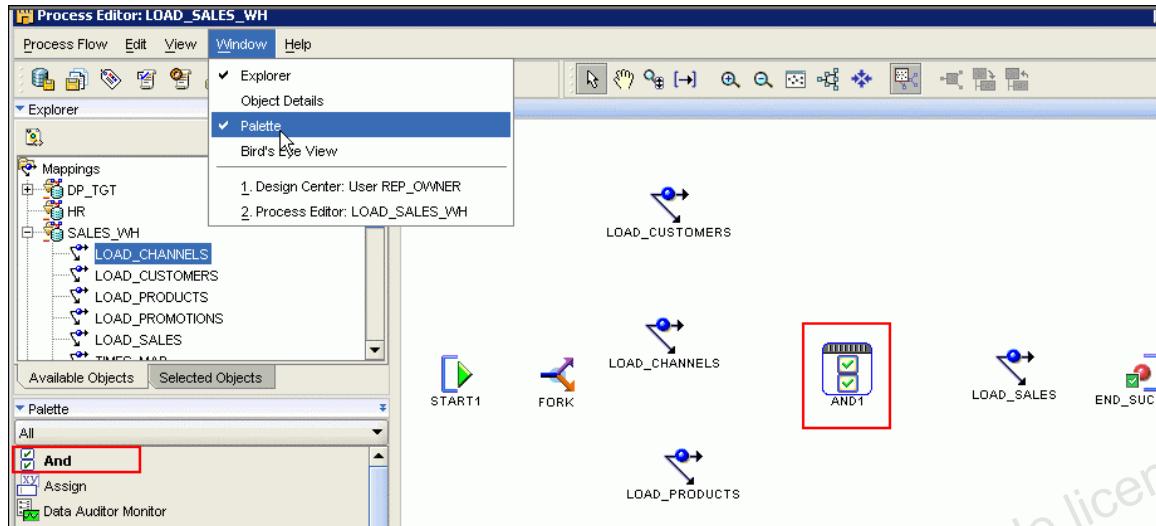
Add an AND Activity

- 8) Drag an AND activity to the canvas.

Answer:

From the Palette, drag the AND activity to the canvas. If the Palette is not visible, from the Window menu, select Palette.

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)

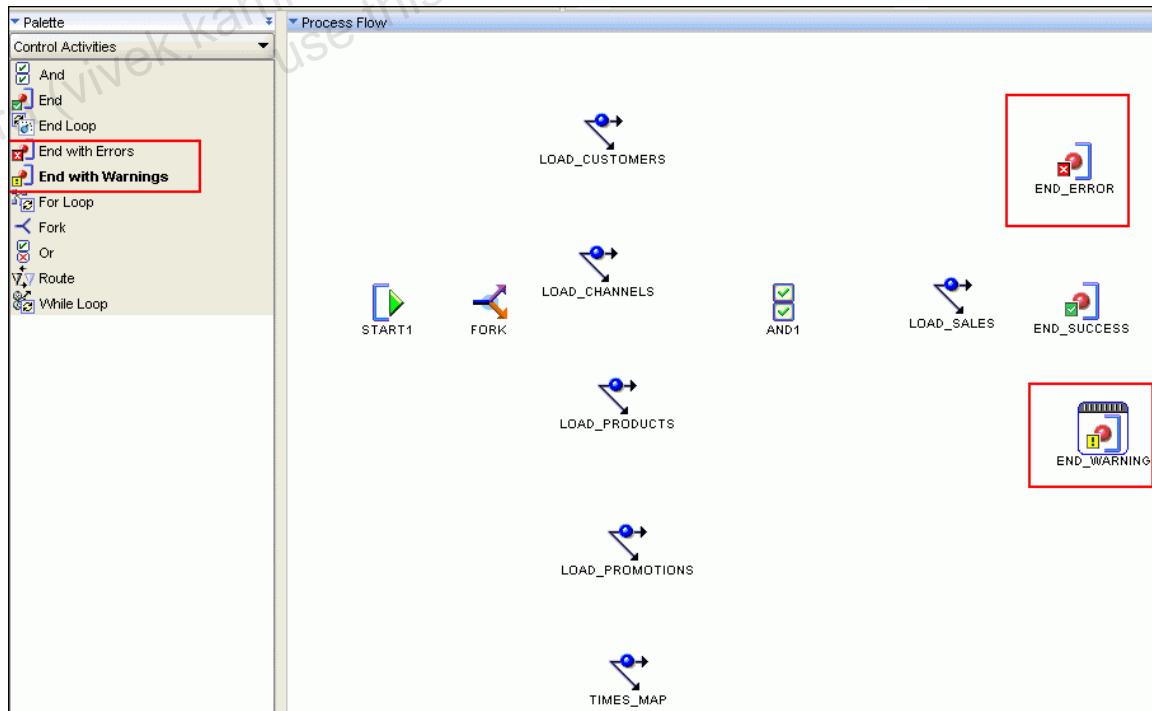


Add Control Activities

- 9) Drag the following activities to the canvas:
- End with Errors
 - End with Warnings

Answer:

On the Process Editor, from the Palette panel, drag the **End with Errors** and **End with Warnings** activities to the canvas as follows:



Add Transitions

**Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow
(continued)**

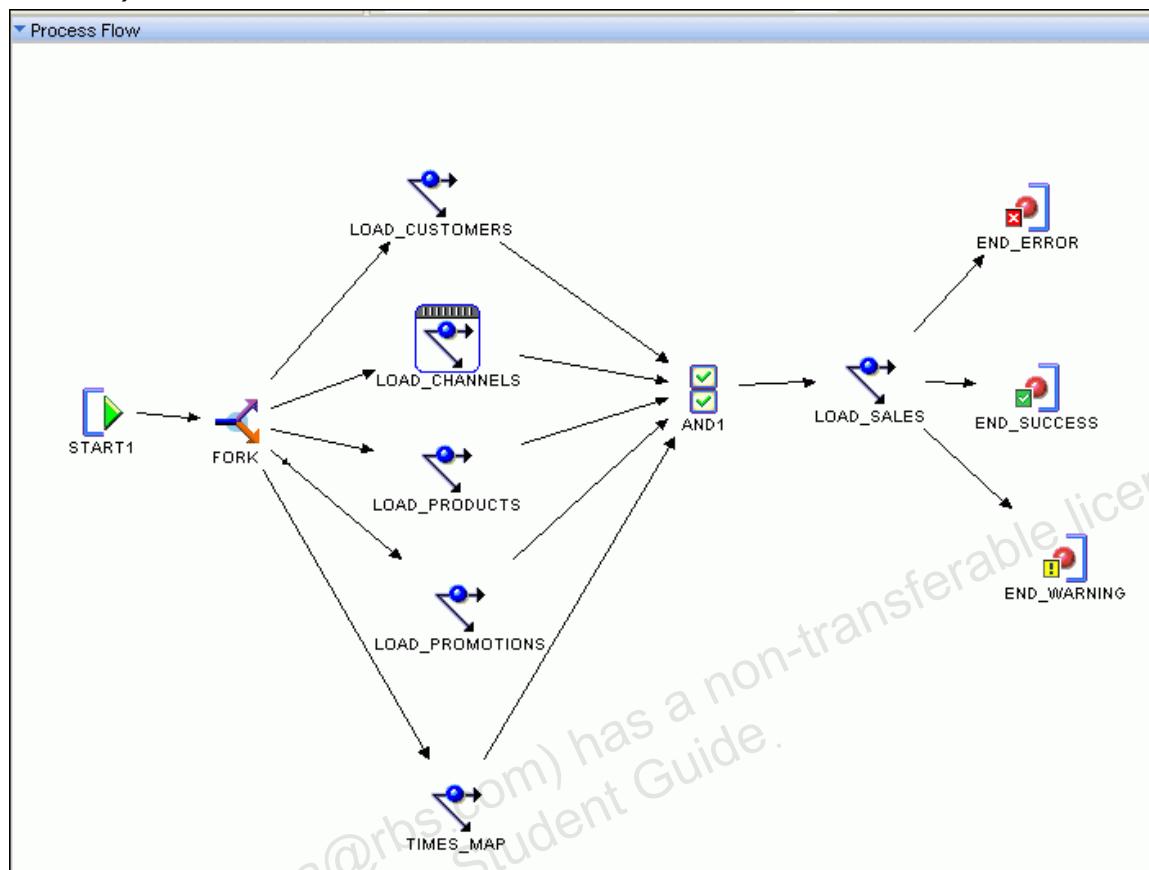
10) Connect the activities using transitions as follows:

FROM	TO
START	FORK
FORK	LOAD_CHANNELS
FORK	LOAD_PROMOTIONS
FORK	LOAD_CUSTOMERS
FORK	LOAD_PRODUCTS
FORK	TIMES_MAP
LOAD_CHANNELS	AND
LOAD_PROMOTIONS	AND
LOAD_CUSTOMERS	AND
LOAD_PRODUCTS	AND
TIMES_MAP	AND
AND	LOAD_SALES
LOAD_SALES	END_SUCCESS
LOAD_SALES	END_ERROR
LOAD_SALES	END_WARNING

Answer:

Place the cursor (it changes to a single-headed arrow) inside an activity that acts as the source (for example, START). Keeping the left mouse button clicked, drag and release the left mouse pointer when you are in the center of the target activity. Draw transitions as shown in the screenshot:

**Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow
(continued)**



- 11) Save your work.

Answer:

In the Process Editor, click the Save All icon on the toolbar. Click Yes in the Warehouse Warning dialog box.

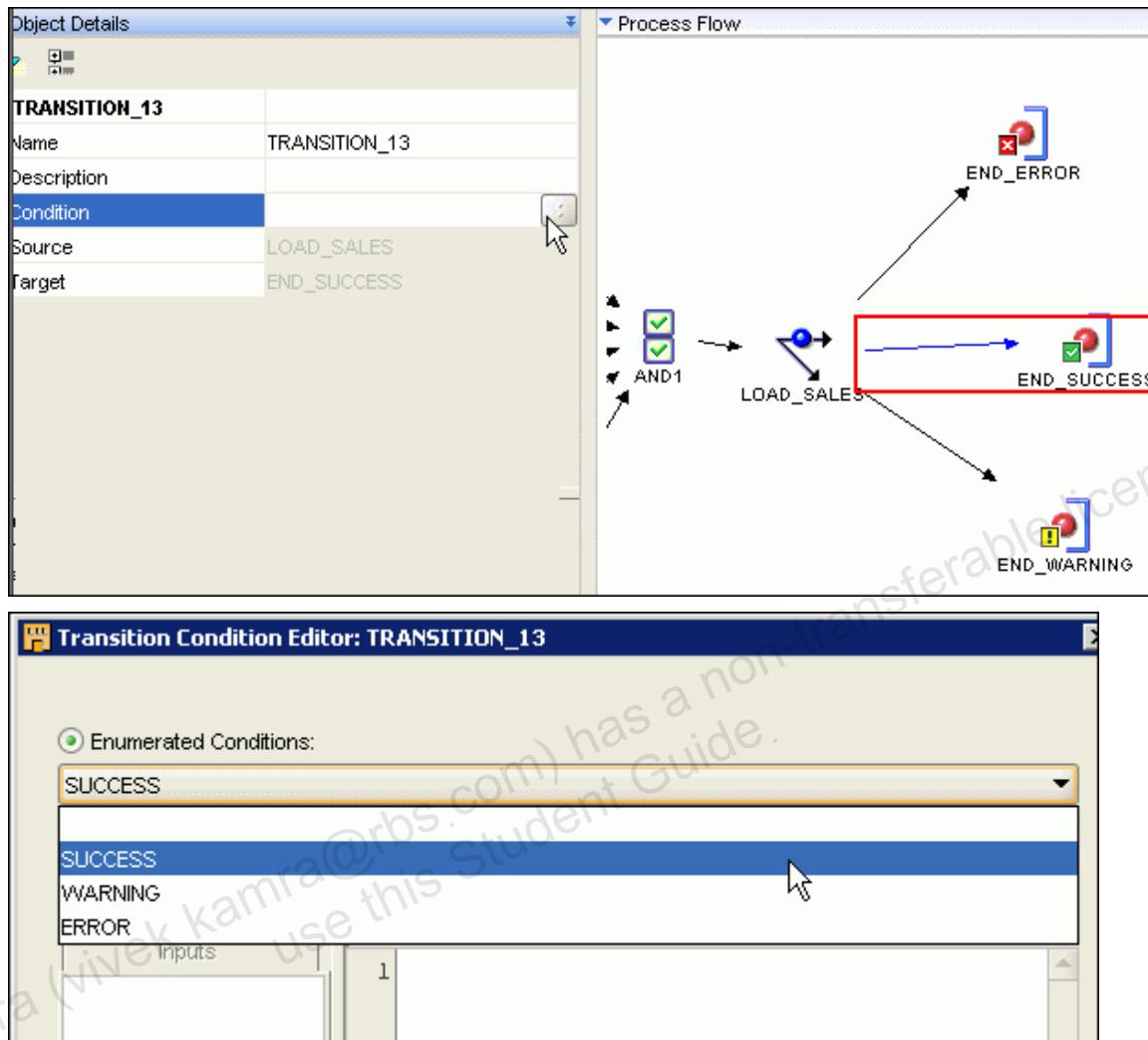
ERROR Handling Logic: Defining Conditions for the Transitions

- 12) Specify the SUCCESS enumerated condition for the SUCCESS transition.

Answer:

Click the transition line from the LOAD_SALES activity to the END_SUCCESS activity. In the Object Details panel, click Condition. Click the : (colon) button. In the Transition Condition Editor, select **SUCCESS** from the Enumerated drop-down list. Click **OK**.

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)



- 13) Specify the ERROR enumerated condition for the ERROR activity.

Answer:

Click the transition line from the LOAD_SALES activity to the END_ERROR activity. In the Object Details panel, click Condition. Click the : (colon) button. In the Transition Condition Editor, select **ERROR** from the Enumerated Conditions drop-down list. Click **OK**.

- 14) Specify the WARNING enumerated condition for the WARNING activity.

Answer:

Click the transition line from the LOAD_SALES activity to the END_WARNING activity. In the Object Details panel, click Condition. Click the : (colon) button. In the Transition Condition Editor, select **WARNING** from the Enumerated Conditions drop-down list. Click **OK**.

- 15) Save your work.

Answer:

Solutions for Practice 9-1: Create the LOAD_SALES_WH Process Flow (continued)

In the Process Editor, click the Save All icon on the toolbar. Click Yes in the Warehouse Warning dialog box.

- 16) Validate the process flow.

Answer:

In the Process Editor, select Validate from the Process Flow menu. The validation results are displayed in the Compilation Results panel.

Compilation Results (0 errors, 0 warnings) Last Updated: Wed May 10 17:09:05 ICT 2006			
Message			
	Code	Message	Validation Details
	<input checked="" type="checkbox"/> Success	VLD-0001: Validation completed successfully.	Validation completed successfully.

- 17) Generate the process flow.

Answer:

In the Process Editor, select Generate from the Process Flow menu. The script is generated and displayed in the Compilation Results window.

Compilation Results (0 errors, 0 warnings) Last Updated: Wed May 10 17:18:07 ICT 2006			
Script			
	Code	Edit	Search
5	<XPDLVersion/>		
6	<Vendor/>		
7	<Created/>		
8	<Description>Filler package for single process generation.</Description>		
9	</PackageHeader>		
10	<TypeDeclarations/>		
11	<Applications>		
12	<Application Id="AD6370266D81D47BA93B6E57FB06B85A2" Name="LOAD_SALES">		
13	<FormalParameters>		
14	<FormalParameter Id="EXIT_CODE" Mode="OUT">		
15	<DataType>		

- 18) Close the Process Editor.

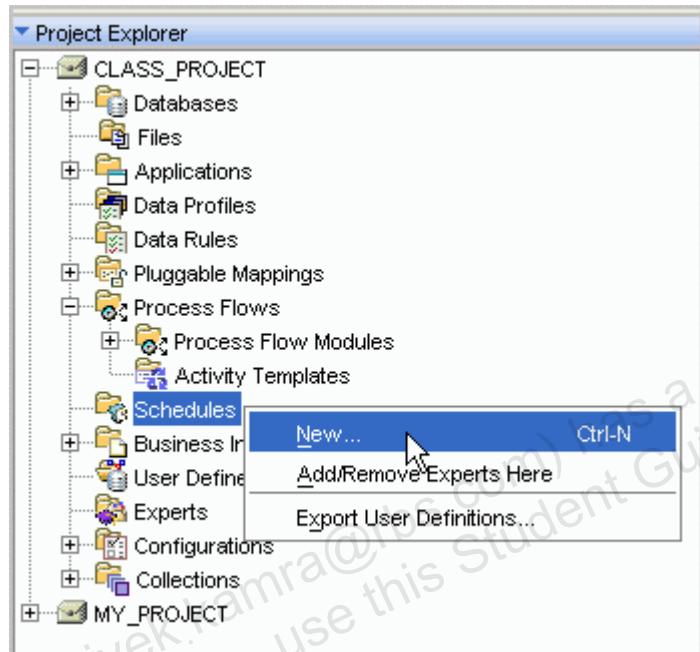
Solutions for Practice 9-2: Create a Schedule

In this practice, you create a schedule that becomes active every two hours.

- 1) Create a Schedule module named SCHED_SALES_WH; the associated location is SALES_WH_LOCATION.

Answer:

1. In the Design Center, expand **CLASS_PROJECT**. Right-click **Schedules**, and select **New**. The Create Module Wizard is launched. Click Next on the Welcome page.



2. On the Name and Description page, enter **SCHED_SALES_WH** as the schedule name, and click **Next**.
 3. On the Connection page, select **SALES_WH_LOCATIONS** from the Locations drop-down list. Click **Next**.
 4. Review the summary information, and click **Finish**.
- 2) Create a schedule named **EVERY_2_HOURS** with the following details:

Time zone for this schedule drop-down select America/Los Angeles

- Date: <enter a date when you want this schedule to start>
- Time: <enter start time> 00:00:00 AM

Choose an end time for this schedule

- Date: <enter a future date>
- Time: 00:00:00 AM

On the “Choose the frequency and repeat interval page,” specify the following:

Frequency: Daily

Repeat every: 1 days

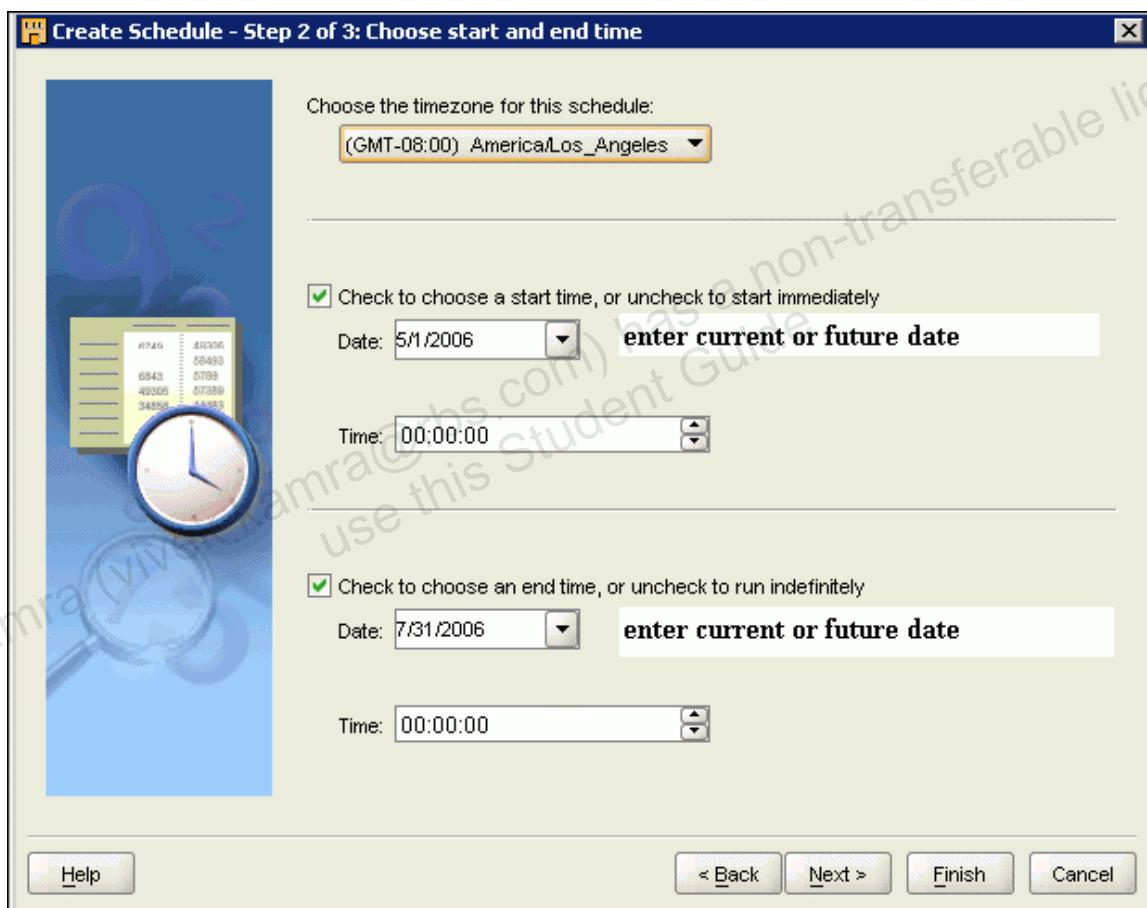
Answer:

1. In the Design Center, right-click SCHED_SALES_WH, and select New. The Create Schedule Wizard is launched. Click Next on the Welcome page.
2. On the Create Schedule Wizard's Name and Description page, enter the following information:

Name: **EVERY_2_HOURS**

Click **Next**. The “Choose start and end time” page is displayed.

- 3) On the “Choose start and end time” page, enter the information as shown in the screenshot:

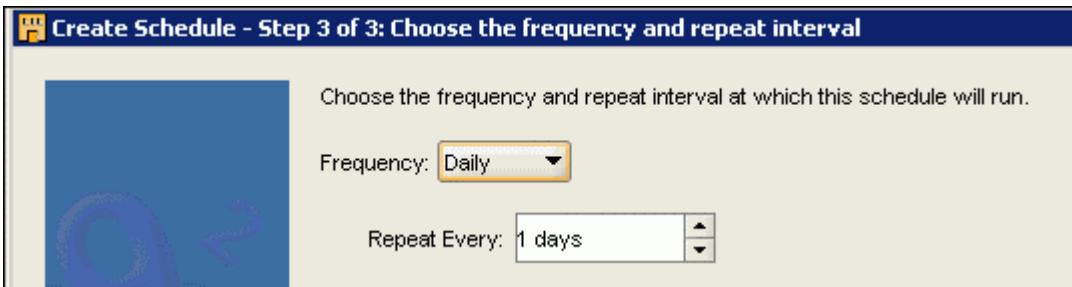


Click **Next**. The “Choose the frequency and repeat interval” page appears.

- 4) On the “Choose the frequency and repeat interval” page, specify the following:

Frequency: **Daily**

Repeat every: **1 days**



Click **Next**.

- 5) Review the Summary page, and click **Finish**.

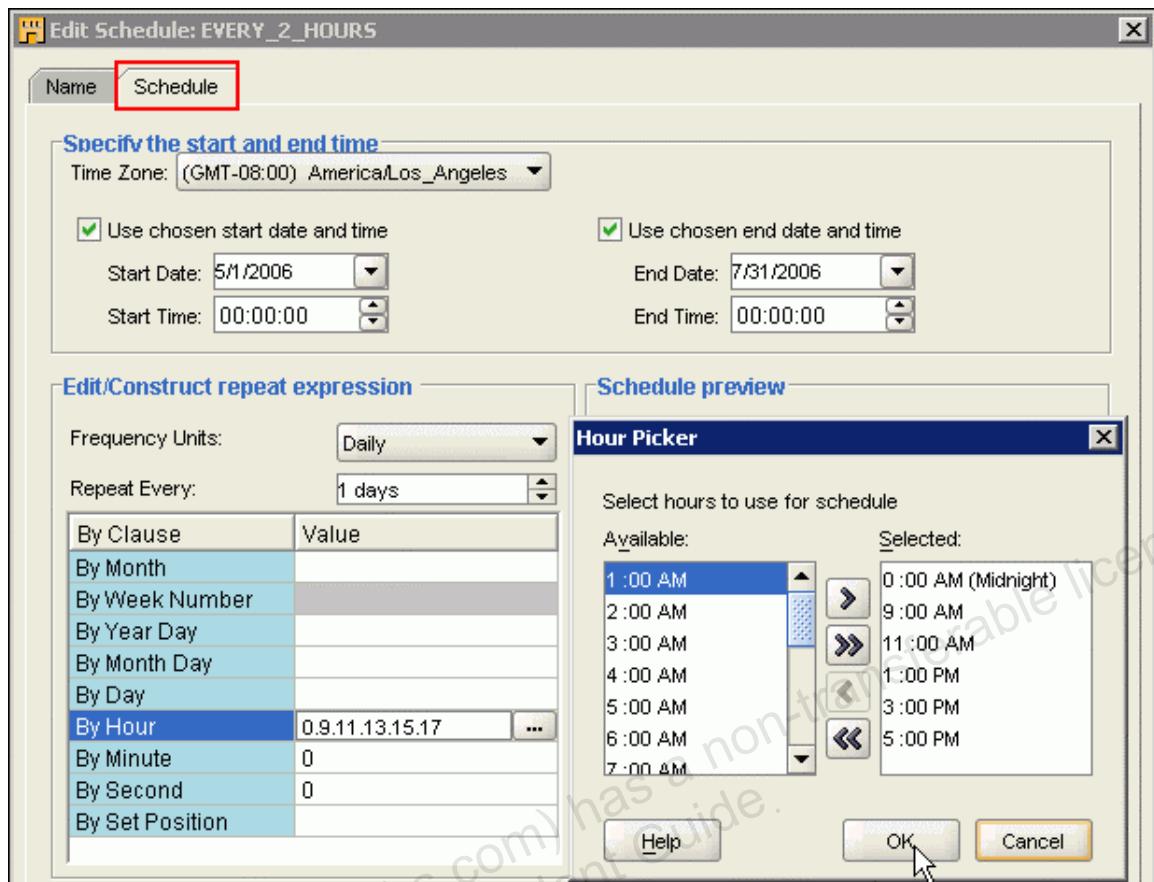
Editing a Schedule

The wizard does not give you all the capabilities to edit the schedule. You get more advanced scheduling capabilities if you edit the schedule.

- 6) Edit the schedule to be able to run at 9, 11, 13, 15, and 17 (5 PM).

Answer:

1. In the Project Explorer in the Design Center, double-click the EVERY_2_HOURS schedule that you created.
2. In the Edit Schedule dialog box, click the Schedule tab.
3. Click the field to the right of By Hour. Click ... (three-dotted button). The Hour Picker pop-up window appears. Move the following values from the Available list to the Selected list: 9: 00 AM, 11: 00 AM, 1: 00 PM, 3: 00 PM, and 5: 00 PM. Click OK.



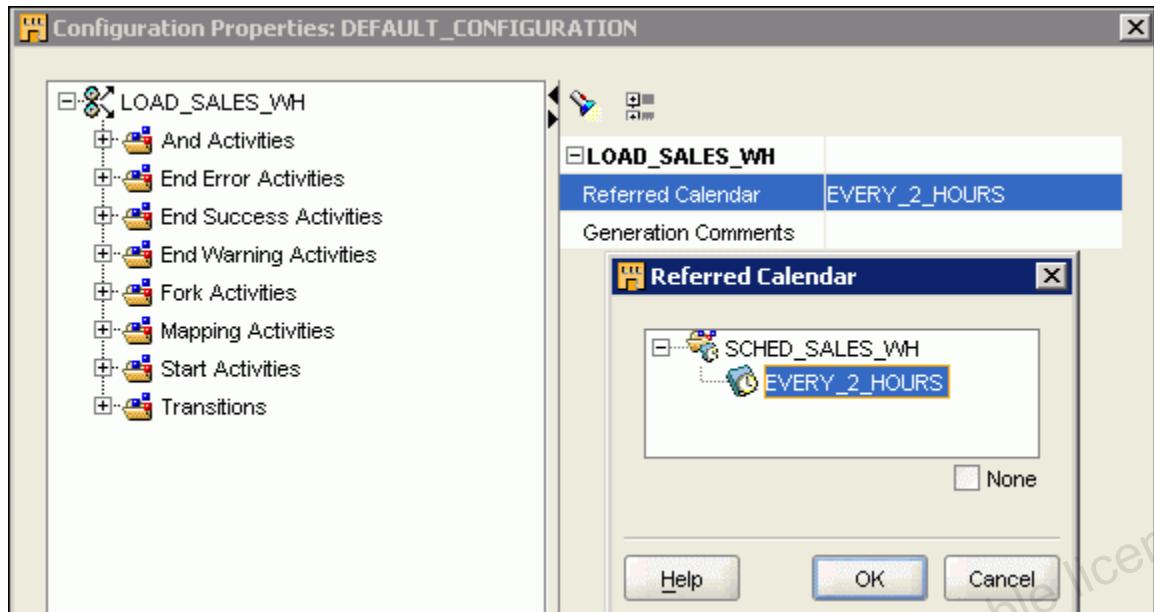
4. Ensure that in the By Minute field and the By Second field the value 0 (zero) is displayed.
5. Click OK in the Edit Schedule window.

Attach a Schedule to a Process Flow

- 7) Specify the name of the EVERY_2_HOURS schedule in the Referred Calendar property of the process flow.

Answer:

1. In the Design Center, in the Project Explorer panel, right-click the LOAD_SALES_WH process flow. Select Configure.
2. In the Configuration Properties dialog box, click the ... (three dotted) button next to the Referred Calendar field. The Referred Calendar dialog box is displayed. In the Referred Calendar dialog box, expand SCHED_SALES_WH, and select EVERY_2_HOURS schedule. Click OK.



3. Click OK in the Configuration Properties dialog box.

Deploy and Execute the Scheduled Job

Note: Do not perform this step at this moment. You can come back to this step after completing Practice 10-1 Loading, Deploying, and Viewing Data, where you deploy the LOAD_SALES_WH process flow. You may continue the steps here but you will not get any results because the LOAD_SALES_WH process flow is not deployed yet.

As the name suggests, schedules run at a scheduled date and time, depending on what you specified.

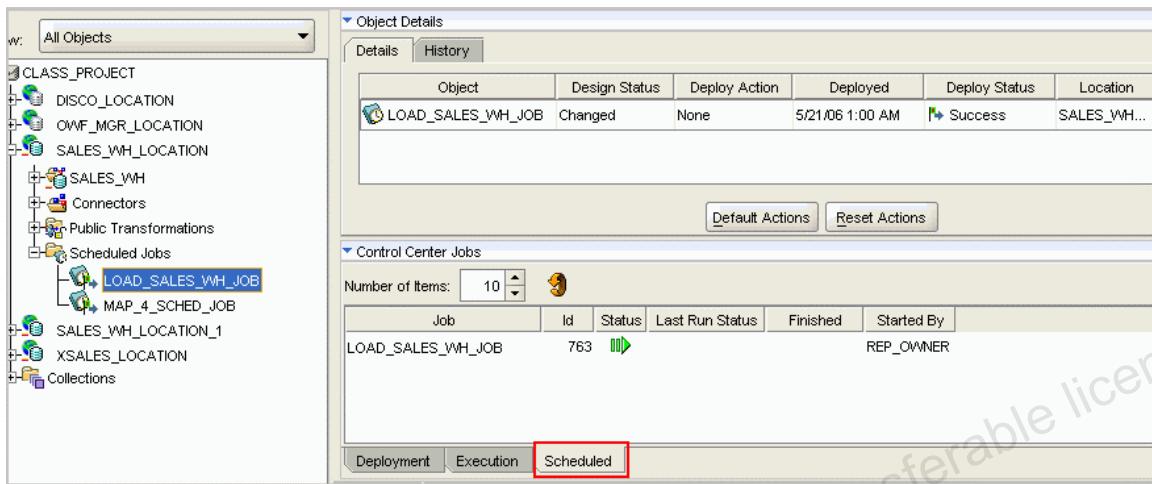
- 8) Examine the scheduled job entry in the Control Center Manager.

Answer:

1. In the Control Center Manager, expand SALES_WH_LOCATION > Scheduled Jobs. You see LOAD_SALES_WH_JOB.

Object	Design Status	Deploy Action	Deployed	Deploy Status
LOAD_SALES_WH_JOB	New	None		Not Deployed

2. Select LOAD_SALES_WH_JOB. In the Object Details panel, click Default Actions. Click the Deploy icon on the toolbar. Wait till deployment completes successfully.
3. Execute LOAD_SALES_WH_JOB.



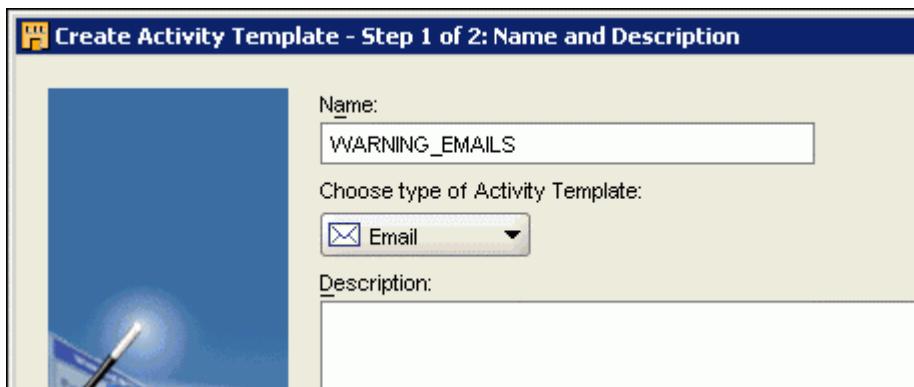
Define Activity Templates

You set up an e-mail activity to be used on a very frequent basis. You create a template that has the values preset; for example, the SMTP server, from address, and to address.

- 9) Create a template named WARNING_EMAILS for an e-mail activity.

Answer:

1. In the Project Explorer, expand Process Flows. Right-click the Activity Templates node, and select New. In the Create Activity Template folder, enter PROCESS_MGT for name. The Create Activity Template Wizard is launched. Click Next on the Welcome page.
2. On the Name and Description page, enter the following values:
 - a. Name: WARNING_EMAILS
 - b. Type of Activity Template: Email



Click Next.

3. On the Parameters page, enter the following information:

Name	Default Value
SMTP Server	rgmareassmtp.company.com
FROM_ADDRESS	bi@company.com
TO_ADDRESS	admin@company.com
IMPORTANCE	High
SUBJECT	WARNING

Click Next.

4. Review the Summary page, and click Finish.

Create a NOTIFICATION Activity

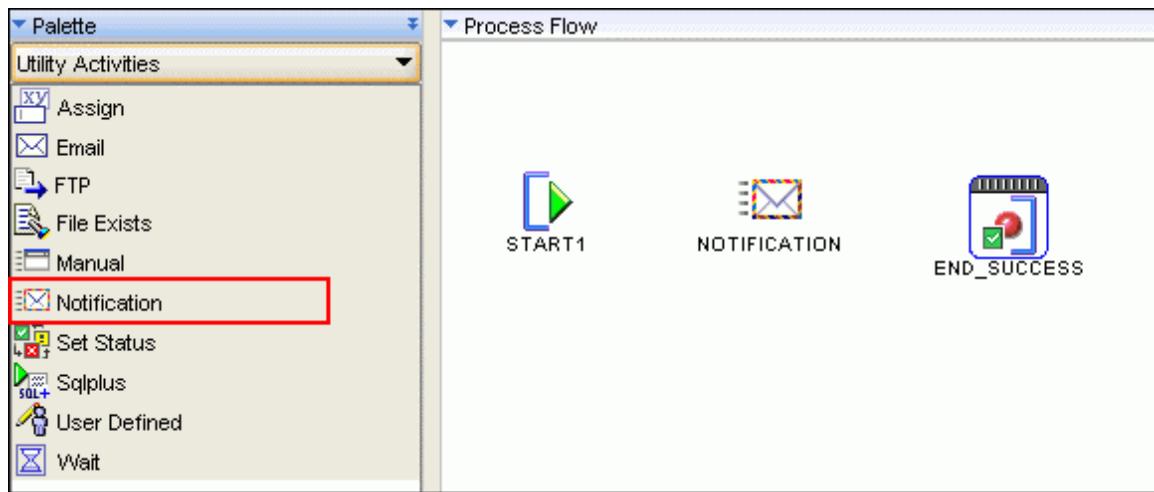
The notification activity enables integration with Oracle Workflow's notification mechanism. For example, users can be sent messages in the form of e-mails to which they have to respond. The response will determine the outcome of the notification.

10) Create a notification named PF_NOTIFICATION under a new process flow package named PK_ACTIVITIES, and enter the following values for the corresponding parameters:

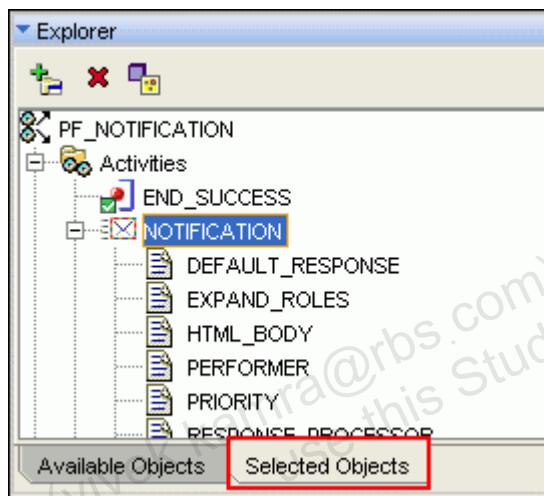
- a. HTML_BODY: '** Specify what you think **'
- b. PERFORMER: 'Mike'
- c. PRIORITY: NORMAL
- d. RESPONSE_TYPE: OK, NOT_OK (These are enumerated values user can choose from.)
- e. DEFAULT_RESPONSE: 'OK'

Answer:

1. In the Project Explorer, expand Process Flows. Right-click PF_SALES_WH, and select New. Enter PK_ACTIVITIES as the name for the process flow package. In the Create Process Flow window, enter PF_NOTIFICATION as the name of the process flow. The Process Editor is launched. (This may take 30 seconds.)
2. In the Palette, select Utility Activities from the drop-down list. Drag a Notification activity to the canvas.

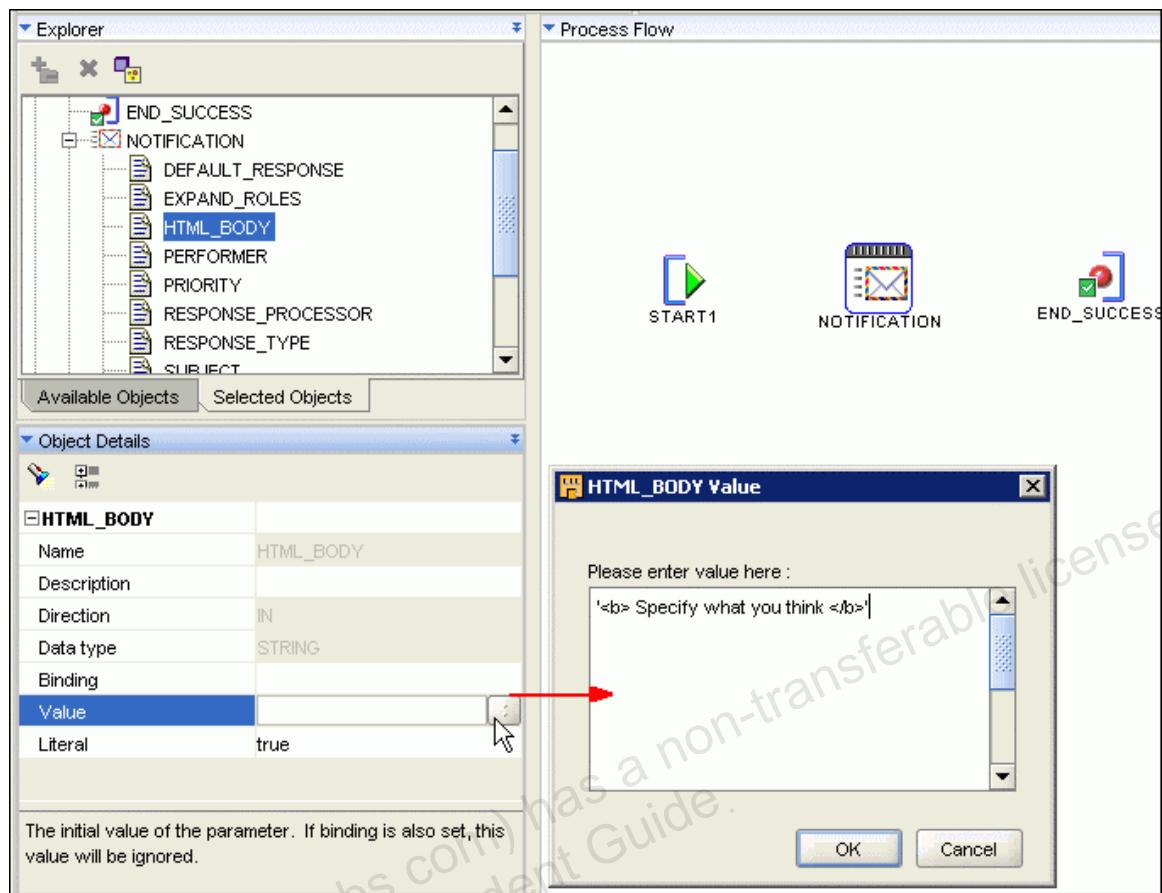


3. In the Explorer panel, click the Selected Objects tab.



4. Expand PF_NOTIFICATION > Activities > NOTIFICATION. Select the parameter in the Explorer, and enter the value in the Object Details panel. Enter values for each parameter as follows:

- HTML_BODY: '**Specify what you think**'



- b. PERFORMER: 'Mike'
- c. RESPONSE_TYPE: OK, NOT_OK (You enter the values here which then appear as enumerated values for the DEFAULT_RESPONSE parameter and the user can choose from the drop-down list.)

Explorer

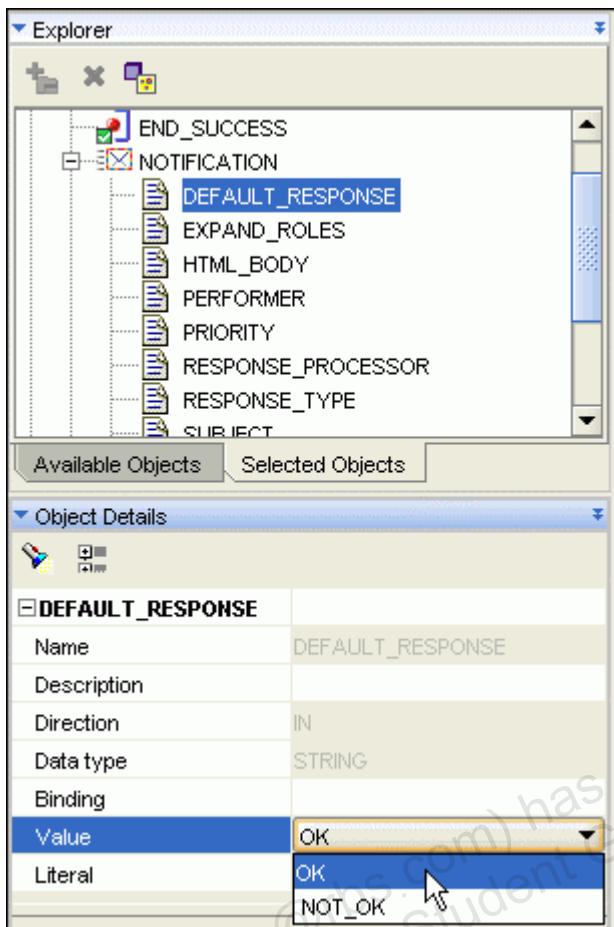
- END_SUCCESS
- NOTIFICATION
 - DEFAULT_RESPONSE
 - EXPAND_ROLES
 - HTML_BODY
 - PERFORMER
 - PRIORITY
 - RESPONSE_PROCESSOR
 - RESPONSE_TYPE
 - SUBJECT

Available Objects Selected Objects

Object Details

RESPONSE_TYPE	
Name	RESPONSE_TYPE
Description	
Direction	IN
Data type	STRING
Binding	
Value	OK, NOT_OK
Literal	true

d. DEFAULT_RESPONSE: 'OK'



- e. SUBJECT: 'Please respond'
- f. TEXT_BODY: 'Specify what you think'

11) Drag an End with Errors activity to the canvas.

Answer:

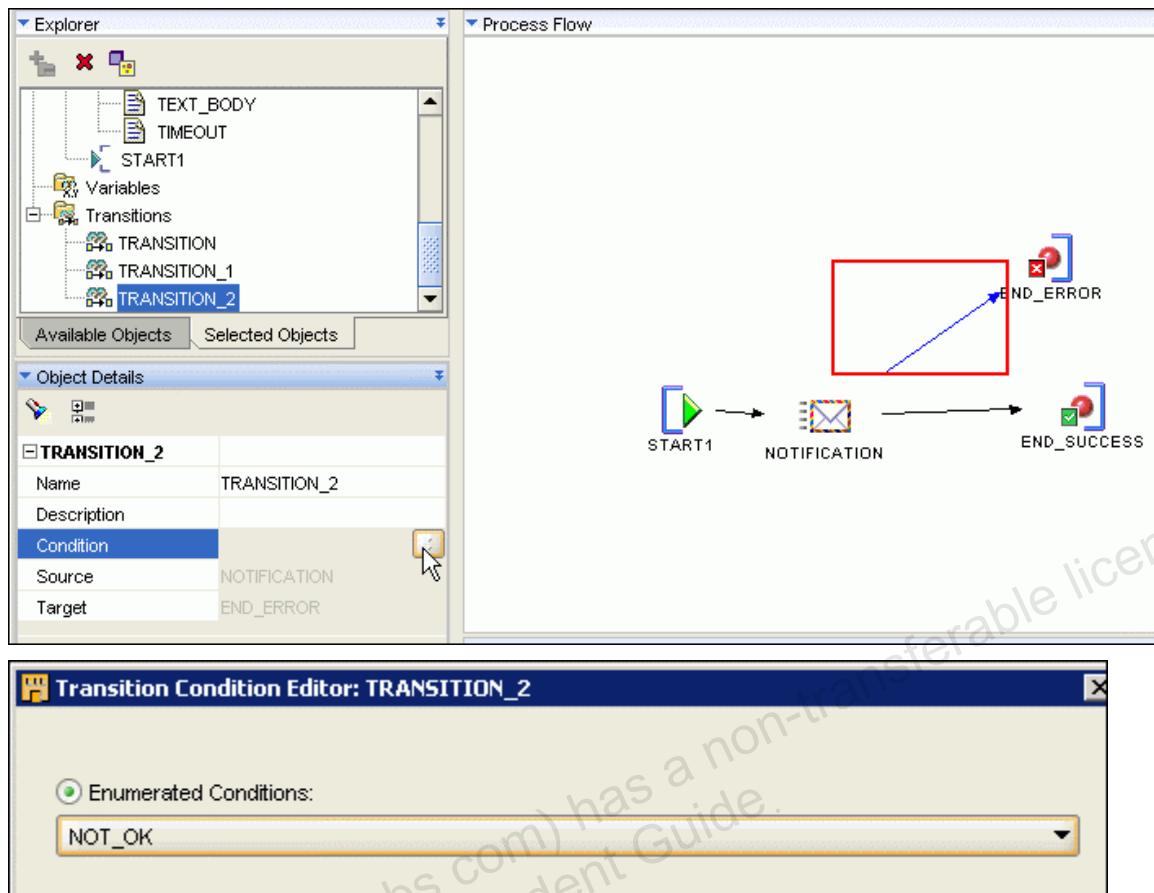
1. In the Palette, select Control Activities from the drop-down list. Drag an End with Errors activity to the canvas.

12) Draw a transition and specify the condition as specified in this table:

From	To	Condition
START	NOTIFICATION	
NOTIFICATION	END_ERROR	NOT_OK
NOTIFICATION	END_SUCCESS	OK

Answer:

1. Draw a transition starting from the START activity to the NOTIFICATION activity. **Note:** There is no Condition for this activity.
2. Draw a transition from the NOTIFICATION activity to the END_ERROR activity. Select the transition starting from NOTIFICATION to the END_ERROR activity. From the Enumerated Conditions drop-down list, select NOT_OK. Click OK.



3. Draw a transition from NOTIFICATION to the END_SUCCESS activity. Select the transition starting from NOTIFICATION to the END_SUCCESS activity. From the Enumerated Conditions drop-down list, select OK. Click OK.
- 13) Save your work.
- 14) Close the Process Editor.

Practice Solutions for Lesson 10

This practice consists of two parts. In Part 1, you deploy the relational and dimensional objects, the ETL mappings, and the process flow. You then execute the process flow, and the data is loaded into the dimensional objects. You use the Data Viewer to view the data that is loaded into the objects.

In Part 2 of this practice, you learn how to manage your metadata using the Dependency Manager in Warehouse Builder. You also use the repository browser, which is the central reporting tool for viewing the data and metadata stored in the Warehouse Builder repository.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 10-1: Deploying, Loading, and Viewing Data

Deploy Relational Objects

Until now you have designed and configured the logical definitions of your target system. Now you learn how to deploy and create the physical instance of your target.

In this practice, you deploy the relational and dimensional objects. You then deploy the ETL mapping. Finally, you deploy and execute the process flow.

- 1) Launch the Control Center Manager.

Answer:

In the Design Center, select **Control Center Manager** from the **Tools** menu. The Control Center Manager is launched.

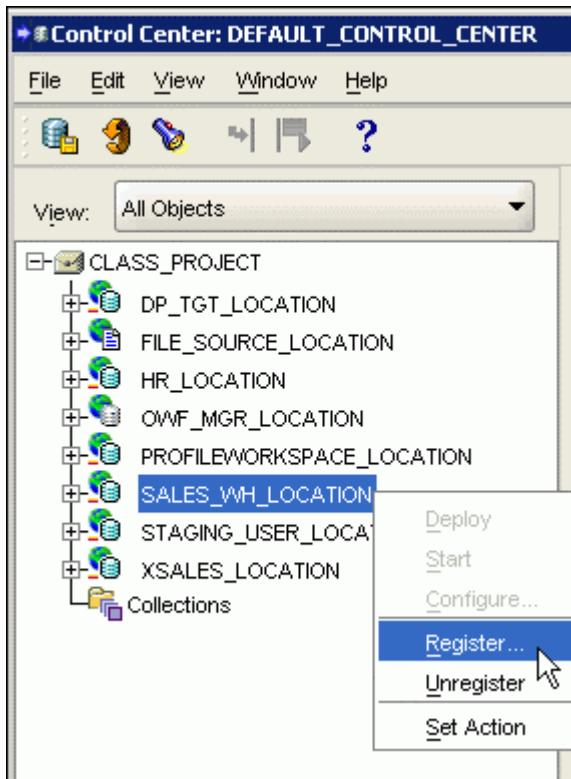
Register the Locations

- 2) Register the following locations:

Location Name	Schema User Name	Password
SALES_WH_LOCATION	SALES_WH	SALES_WH
XSALES_LOCATION	XSALES	XSALES
OWF_MGR_LOCATION	OWF_MGR	OWF_MGR

Answer:

1. In the Control Center, from the navigation tree on the left, right-click SALES_WH_LOCATION, and select **Register**.



2. The Edit Oracle Database Location dialog box appears. Review the details, enter the password as SALES_WH, and click Test Connection. If the connection is successful, click OK.

*** Edit Oracle Database Location: SALES_WH_LOCATION**

This location has not been registered. Please complete the location parameters.

Name:	SALES_WH_LOCATION
Description:	
Type:	HOST:PORT:SERVICE
User Name:	SALES_WH
Password:	*****
Host:	localhost
Port:	1521
Service Name:	orcl
<input type="checkbox"/> Use Global Name:	
Schema:	SALES_WH
Version:	10.2
<input type="button" value="Test Connection"/>	
Test Results: Successful!	

3. Repeat steps 1 and 2 for XSALES_LOCATION. The password is XSALES.
4. Repeat steps 1 and 2 for OWF_MGR_LOCATION. The password is OWF_MGR.

You are now ready to deploy the objects.

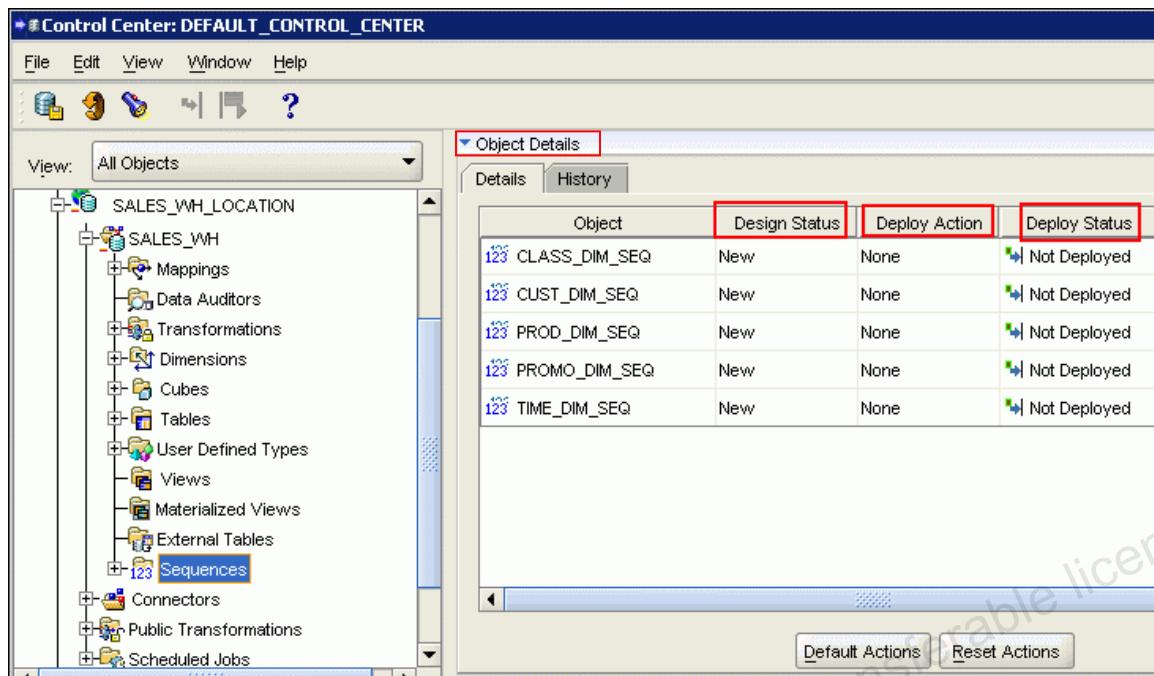
Deploy Sequences

- 3) Deploy the sequences in the SALES_WH target module.

Answer:

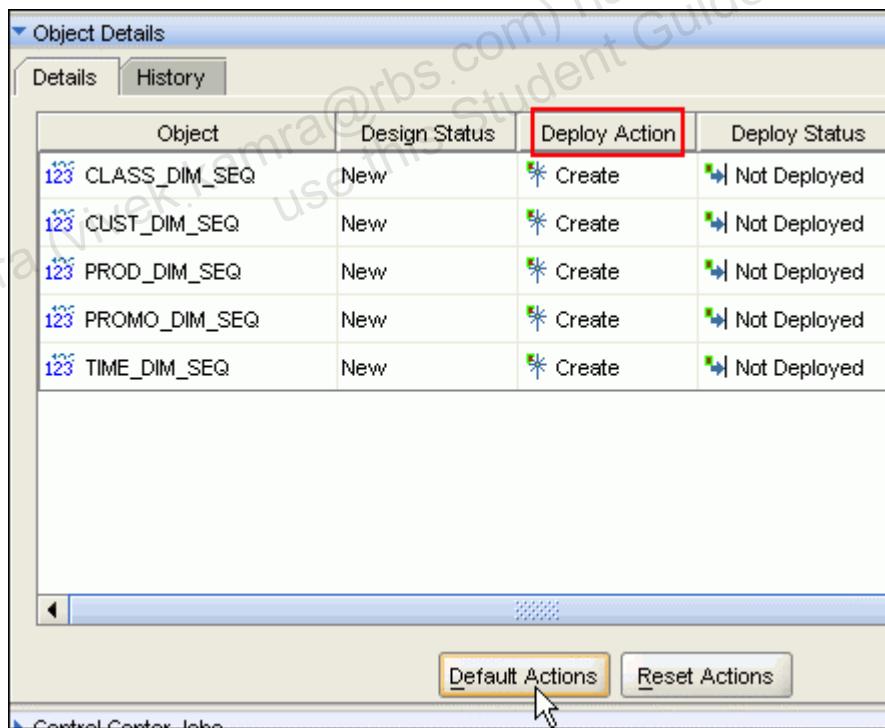
1. In the Control Center, in the left navigation tree, expand **SALES_WH_LOCATION > SALES_WH**. Select **Sequences**. On the right, in the Object Details panel, you see the sequences listed.

Note the values in the Design Status, Deploy Action, and Deploy Status columns.

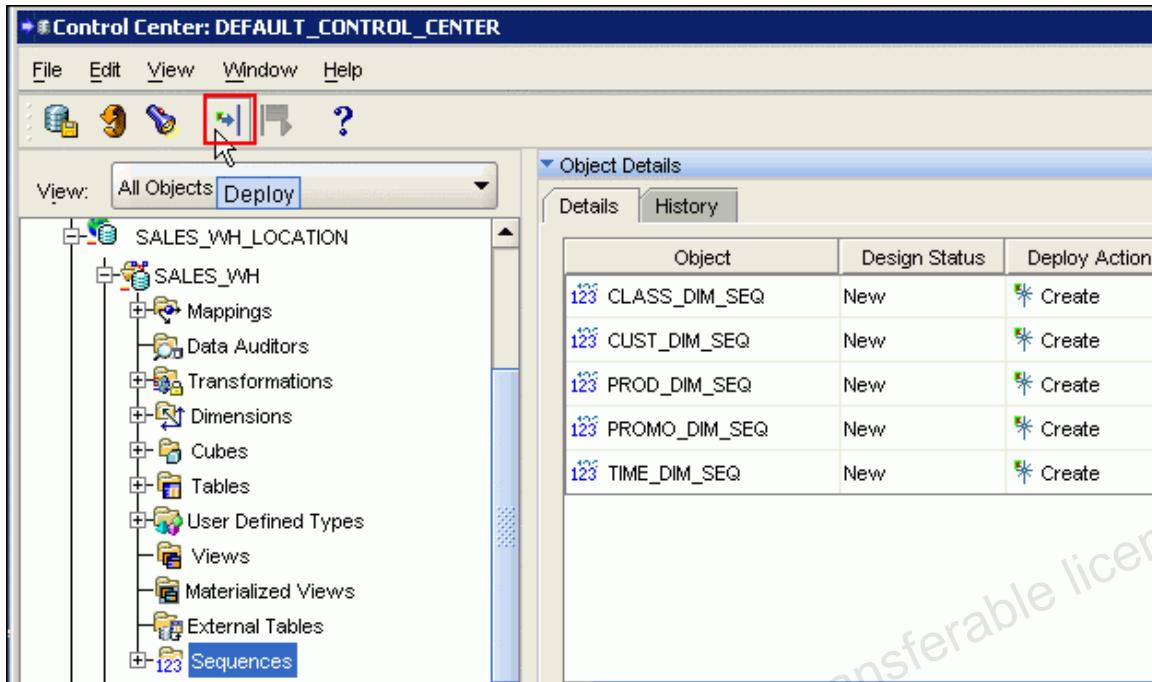


2. Click Default Actions.

Note: Clicking the Default Action button changes deploy action from None to Create.



3. Click the Deploy button on the toolbar.

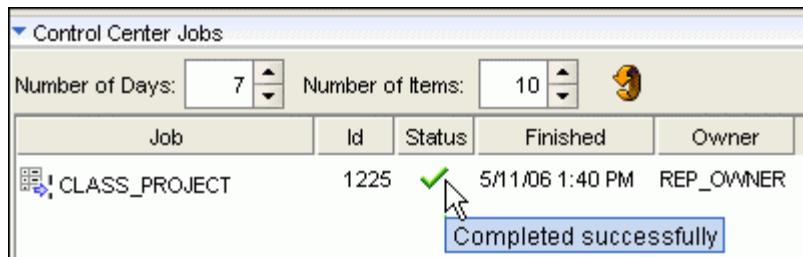


- Monitor the progress in the Control Center Jobs panel. The Deployment tabbed page is displayed. In the Control Center Jobs panel, monitor the deployment progress. The deployment progress moves from **Generation** to **Running** to **Completed Successfully**.

The screenshots show the Control Center Jobs panel with "Number of Days: 7" and "Number of Items: 10". The "Deployment" tab is selected. The table shows three jobs:

Job	Id	Status	Finished	Owner
CLASS_PROJECT	1201			REP_OWNER
CLASS_PROJECT	1081		5/5/06 7:25 PM	REP_OWNER
CLASS_PROJECT	1038		5/4/06 8:37 PM	REP_OWNER

In the second screenshot, the status for job 1038 has changed to "Running".



5. Note the value in the Deploy Status column. It displays Success against each object that has been deployed successfully.

Object Details			
Object	Design Status	Deploy Action	Deploy Status
123 CLASS_DIM_SEQ	Unchanged	None	Success
123 CUST_DIM_SEQ	Unchanged	None	Success
123 PROD_DIM_SEQ	Unchanged	None	Success
123 PROMO_DIM_SEQ	Unchanged	None	Success
123 TIME_DIM_SEQ	Unchanged	None	Success

You have deployed the Sequence objects. You now deploy the Tables objects.

Deploy Tables

- 4) Deploy the Tables in the SALES_WH target module.

Answer:

1. In the Control Center, in the navigation tree, expand **SALES_WH_LOCATION** > **SALES_WH**. Select **Tables**. In the Object Details panel, the tables are listed. Note the values in the Design Status column, Deploy Action column, and Deploy Status column.
2. Click **Default Actions**. **Note:** Clicking Default Action changes value in the Deploy Action column from None to Create.
3. Click the **Deploy** button on the toolbar. In the Control Center Jobs panel, monitor the deployment progress. The deployment progress moves from Generate to Run to Successful. Wait till the Deploy Status column displays Success.

You have deployed the Tables objects. You now deploy the dimensional objects. Before you can deploy the cube, you should deploy the related dimensions.

Deploy Dimensional Objects

- 5) Deploy the dimensions in the SALES_WH target module.

Answer:

1. In the Control Center, in the navigation tree, expand SALES_WH_LOCATION > SALES_WH. Select **Dimensions**. In the Object Details panel, the dimensions are listed. Note the values in the Design Status column, Deploy Action column, and Deploy Status column.
2. Click **Default Actions**. **Note:** Clicking Default Action changes value in the Deploy Action column from None to Create.
3. Click the **Deploy** button on the toolbar. In the Control Center Jobs panel, monitor the deployment progress. The deployment progress moves from Generate to Run to Successful. Wait till the Deploy Status column displays Success.

Deploy the Cube

- 6) Deploy the cube in the SALES_WH target module.

Answer:

1. In the Control Center, in the navigation tree, expand SALES_WH_LOCATION > SALES_WH. Select **Cubes**. In the Object Details panel, the cubes are listed. Note the values in the Design Status column, Deploy Action column, and Deploy Status column.
2. Click **Default Actions**. **Note:** Clicking Default Action changes value in the Deploy Action column from None to Create.
3. Click the **Deploy** button on the toolbar. In the Control Center Jobs panel, monitor the deployment progress. The deployment progress moves from Generate to Run to Successful. Wait till the Deploy Status column displays Success.

Deploy the Mappings

- 7) Deploy the mappings in the SALES_WH target module.

Answer:

1. In the Control Center, in the navigation tree, expand SALES_WH_LOCATION > SALES_WH. Select **Mappings**. In the Object Details panel, the mappings are listed. Note the values in the Design Status column, Deploy Action column, and Deploy Status column.
2. Click **Default Actions**. **Note:** Clicking Default Action changes value in the Deploy Action column from None to Create.
3. Click **Deploy** button on the toolbar. In the Control Center Jobs panel, monitor the deployment progress. The deployment progress moves from Generate to Run to Successful. Wait till the Deploy Status column displays Success.

Deploy the Process Flow

- 8) Deploy the LOAD_SALES_WH process flow. Expand OWF_MGR_LOCATION to find the LOAD_SALES_WH process flow.

Answer:

- In the Control Center, in the navigation tree, expand **OWF_MGR_LOCATION** > **PF_SALES_WH**. Select **PK_SALES**. In the Object Details panel, the process flow package is listed.

Note: When you deploy a process flow, all process flows listed in the package are deployed.

- Click **Default Actions**.
- Click the **Deploy** button on the toolbar. In the Control Center Jobs panel, monitor the deployment progress. The deployment progress moves from Generate to Run to Successful. Wait till the Deploy Status column displays Success.

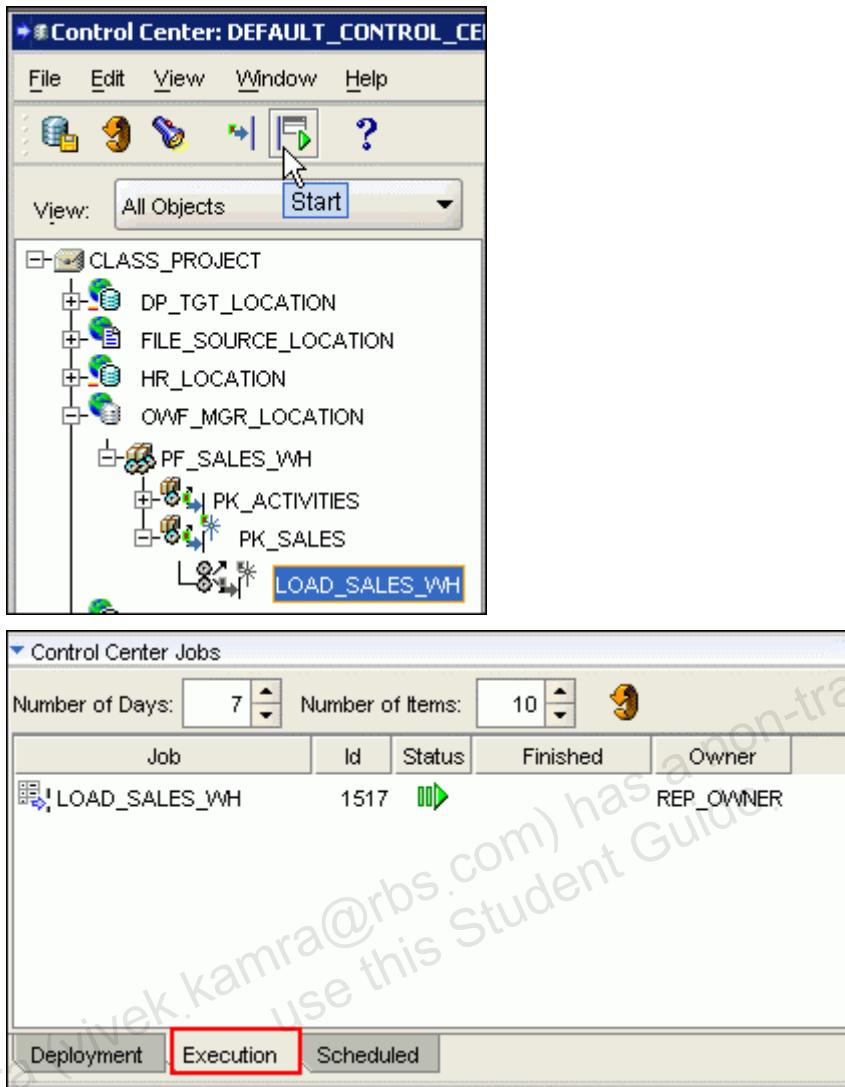
The screenshot shows the Oracle Warehouse Builder Control Center interface. On the left, the navigation tree displays various project locations and objects. Under the 'CLASS_PROJECT' node, there are several locations like 'DP_TGT_LOCATION', 'FILE_SOURCE_LOCATION', 'HR_LOCATION', 'OWF_MGR_LOCATION', and 'PF_SALES_WH'. Within 'PF_SALES_WH', there are 'PK_ACTIVITIES' and 'PK_SALES'. The 'LOAD_SALES_WH' object is also visible under 'PF_SALES_WH'. On the right, the 'Object Details' panel is open, showing a table with one row for 'PK_SALES'. The table columns are 'Object', 'Design Status', 'Deploy Action', 'Deployed', and 'Deploy Status'. The 'Object' column shows 'PK_SALES', 'Design Status' is 'Unchanged', 'Deploy Action' is 'None', 'Deployed' is '5/11/06 3:47 PM', and 'Deploy Status' is 'Success'. Below the table are 'Default Actions' and 'Reset Actions' buttons. At the bottom, the 'Control Center Jobs' panel is shown, with settings for 'Number of Days' (7) and 'Number of Items' (10), and a table listing a single job named 'CLASS_PROJECT' with ID 1449, status 'Finished', and owner 'REP_OWNER'.

Execute the Process Flow

- Execute the **LOAD_SALES_WH** process flow.

Answer:

- Select the **LOAD_SALES_WH** process flow.
- To execute a process flow, click the Start button on the toolbar. **Note:** On clicking the Start button, the Control Center Jobs panel in the lower section switches to the Execution tab.



3. In the Control Center Jobs panel, the Execution tabbed page is displayed. Monitor the execution till it has completed successfully.

Note: The executions are performed asynchronously, which means that you can close the Control Center and OWB client while your process flow or mapping is being executed.

- 10) View the execution details.

Answer:

7. In the Control Center Jobs panel, ensure that the Execution tabbed page is displayed. Double-click LOAD_SALES_WH in the Job column.
8. Click the Execution Results tab and monitor the row activity. **Note:** You get a count of how many rows were inserted in respective dimensions and the cube.

- 11) Close the Control Center Manager.

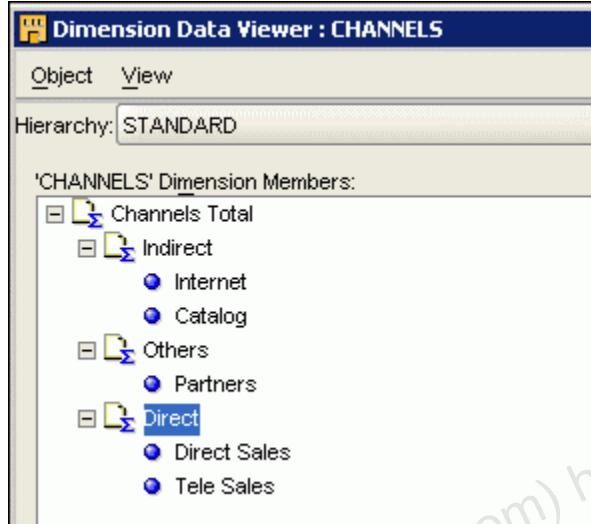
Viewing Data Using Data Viewer

Using the data viewer, you can view the data loaded into the dimensions and the cube from inside the OWB design client environment.

12) View the CHANNELS dimension data.

Answer:

1. In the Project Explorer in the Design Center, expand SALES_WH > Dimensions. Right-click CHANNELS, and select Data. The Dimension Data Viewer is launched and displays the data.



2. Repeat step 1 to view data for the other dimension.

13) Close the Dimension Data Viewer.

14) View the SALES cube data.

Answer:

1. In the Project Explorer, expand SALES_WH > Cubes. Right-click SALES, and select Data. The Cube Data Viewer is launched.

15) Close the Cube Data Viewer. .

Solutions for Practice 10-2: Metadata Management

The Warehouse Builder Dependency Manager enables you to determine the impact of the changes made to the object definitions or the metadata in the Warehouse Builder repository. You also learn how to use the Oracle Warehouse Builder Repository Browser, an HTML-based interface to view and generate reports on all repository metadata objects and the relationships between those objects.

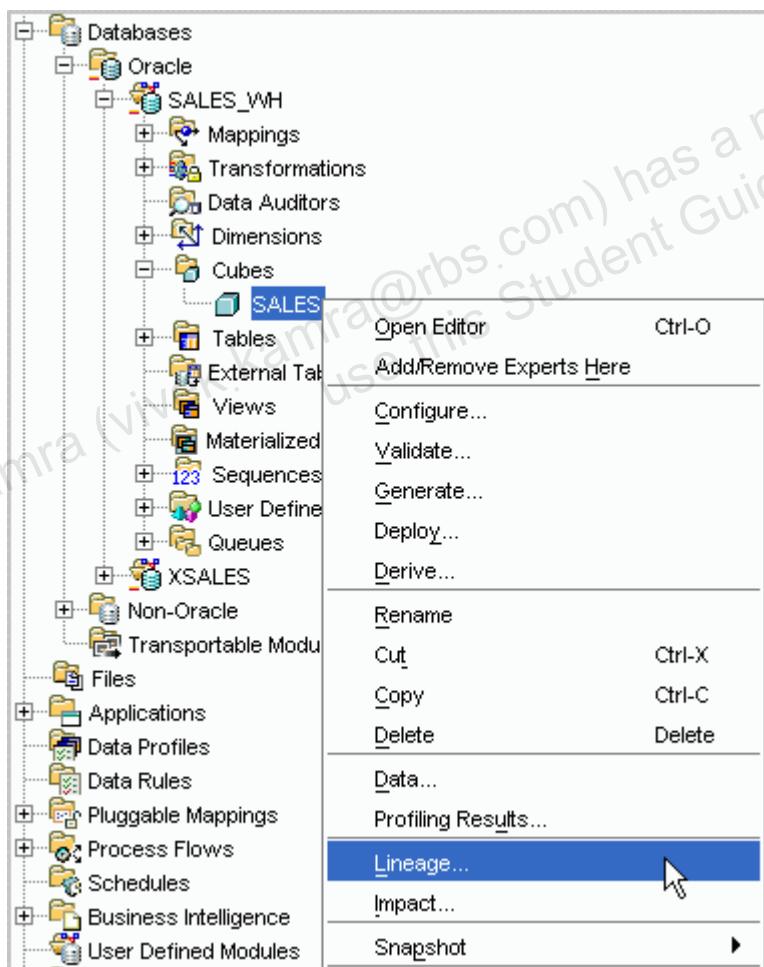
The Metadata Dependency Manager generates lineage and impact diagrams for any data object. A lineage diagram traces the process flows for an object back to the data source and displays all objects along that path. An impact diagram identifies all the objects that are derived from selected object.

Lineage Analysis for the SALES Cube

- 1) Launch the Metadata Dependency Manager for a lineage analysis of the SALES cube.

Answer:

In the Project Explorer, expand **SALES_WH > Cubes**. Right-click **SALES**, and select **Lineage**. The Metadata Dependency Manager is launched.

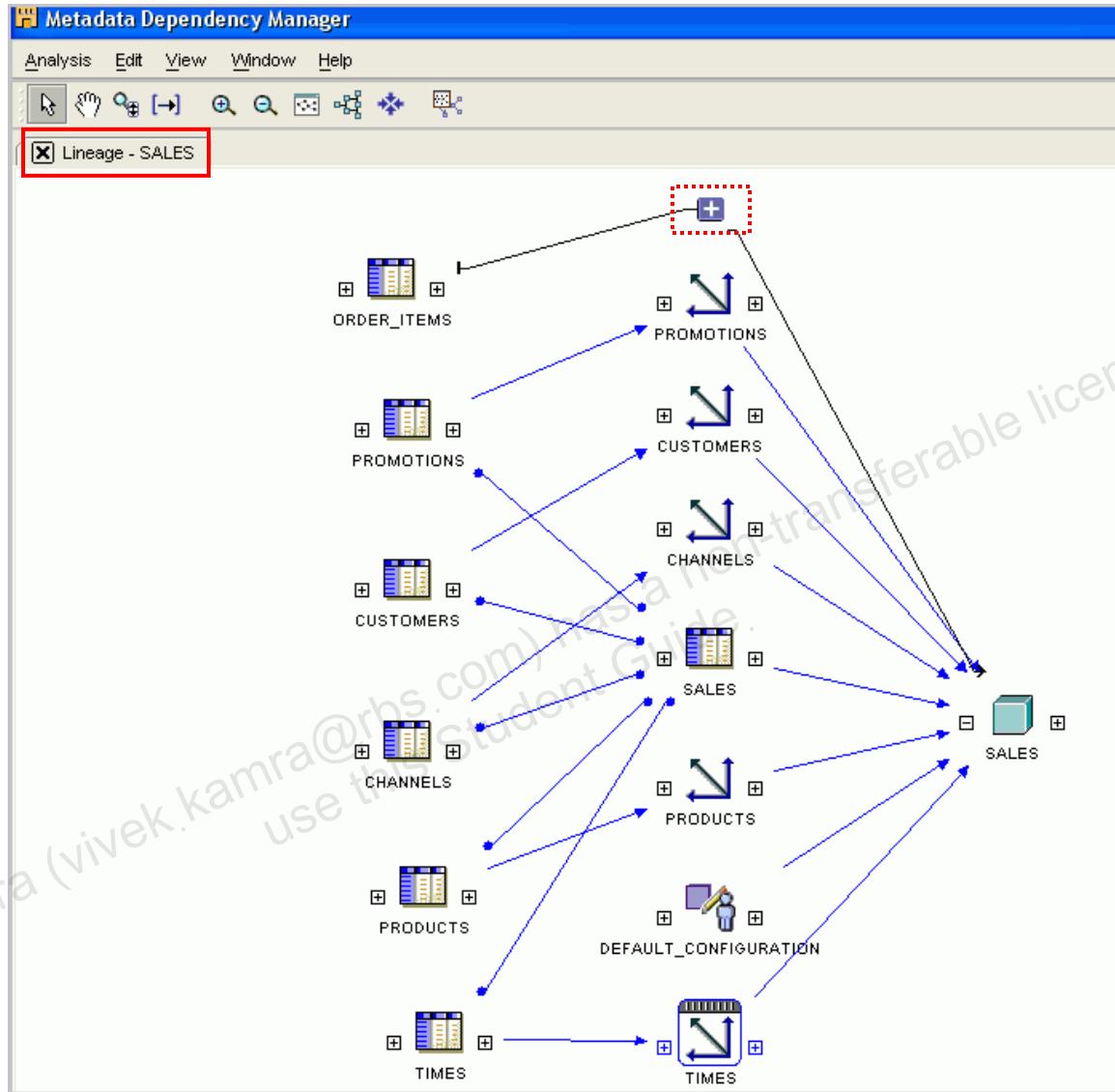


- 2) Find out where all the information in the SALES cube comes from.

Answer:

Solutions for Practice 10-2: Metadata Management (continued)

1. Click + (plus sign) on the left of the SALES cube. You then get this expanded diagram. The lineage diagram for the SALES cube includes the five dimensions, the SALES cube, and the default configuration.

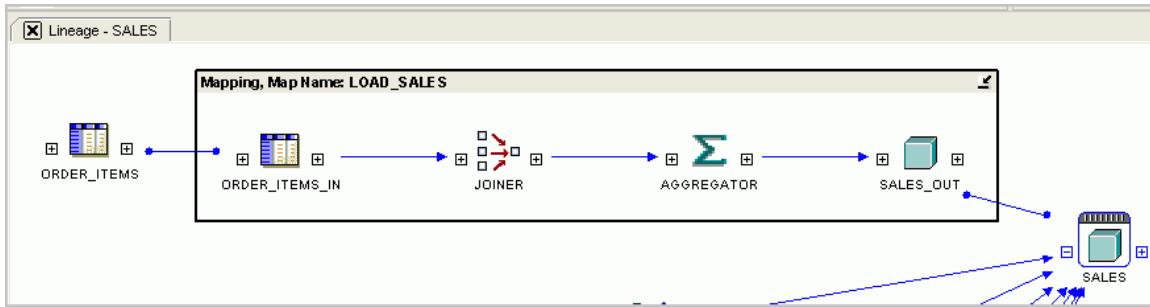


- 3) What is the connection between the ORDER_ITEMS table and the SALES cube?

Answer:

1. Double-click the + (plus) symbol (represents a collapsed mapping piece) that is in between the ORDER_ITEMS table and the SALES cube. The Metadata Dependency Manager not only calculates the impact and lineage on the level of data objects, but also on the level of individual attributes.

Solutions for Practice 10-2: Metadata Management (continued)

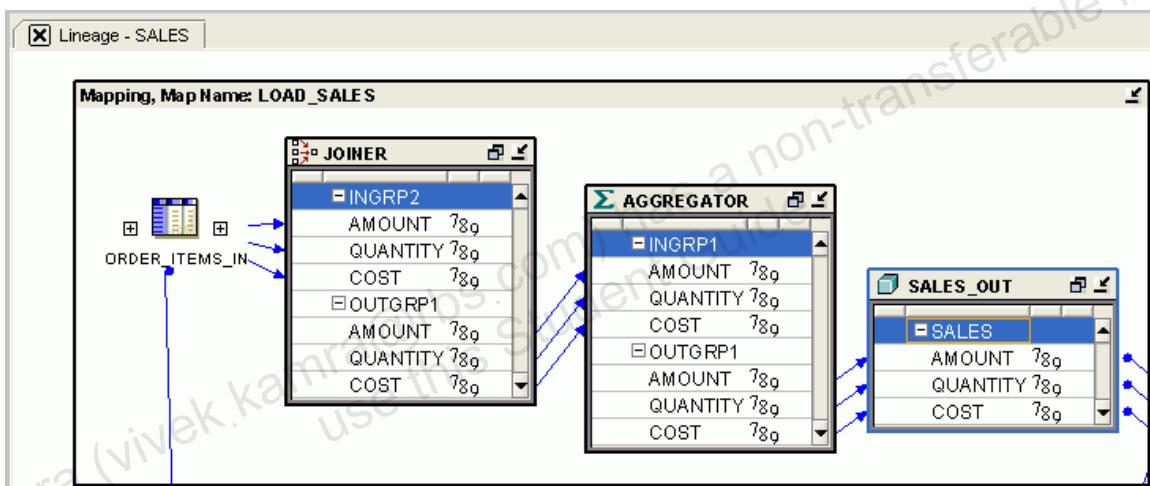


Drill Down to Attribute Level Impact Analysis

- 4) Find out which attributes are being processed by the aggregator operator.

Answer:

Double-click the AGGREGATOR operator, JOINER, and SALES_OUT inside the mapping box to view the input and the output attributes.



- 5) Close the Metadata Dependency Manager window.

Answer:

In the Metadata Dependency Manager window, select Close from the Analysis menu.

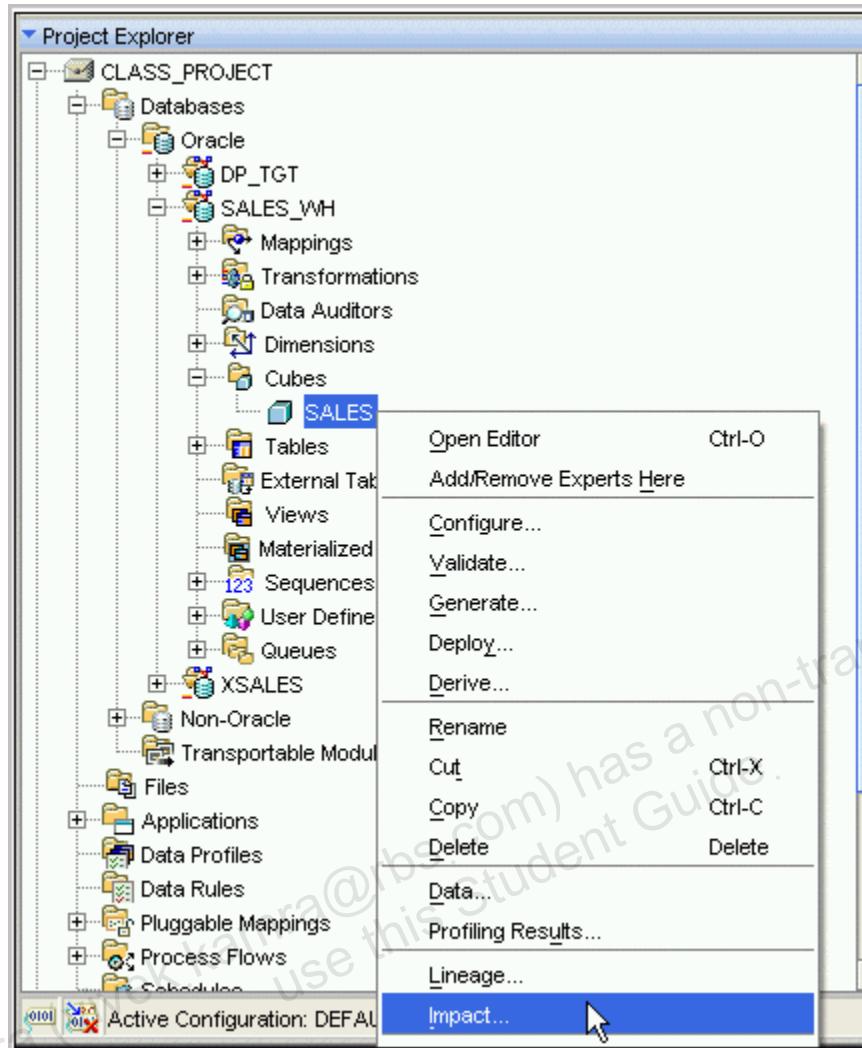
Impact Analysis for the SALES Cube

- 6) Launch the Metadata Dependency Manager for an impact analysis of the SALES cube.

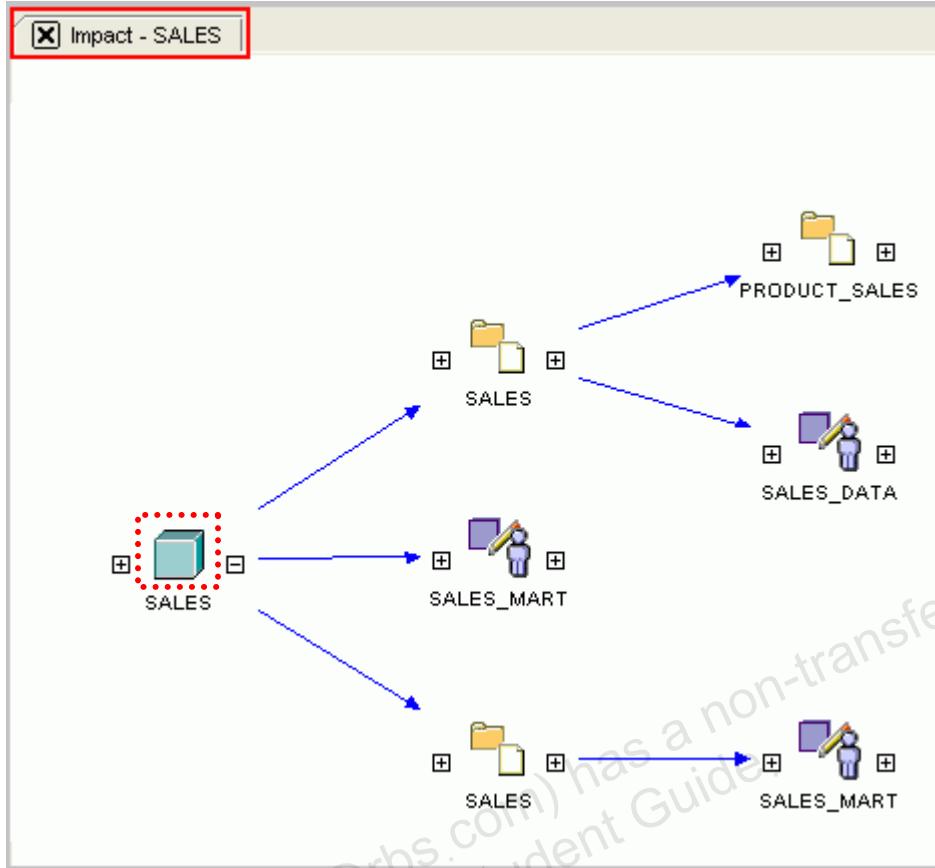
Answer:

1. In the Project Explorer, expand **SALES_WH > Cubes**. Right-click **SALES**, and select **Impact**. The Metadata Dependency Manager is launched.

Solutions for Practice 10-2: Metadata Management (continued)



Solutions for Practice 10-2: Metadata Management (continued)



2. The Impact Analysis Diagram shows an impact diagram of the SALES cube, which includes the Business Intelligence objects that you created. If you are not able to see the same output as on the screen, click + (plus sign) on the right of the SALES cube operator.

- 7) Close the Metadata Dependency Manager.

Administering the Warehouse Using Repository Browser

Start the OWB Browser Listener

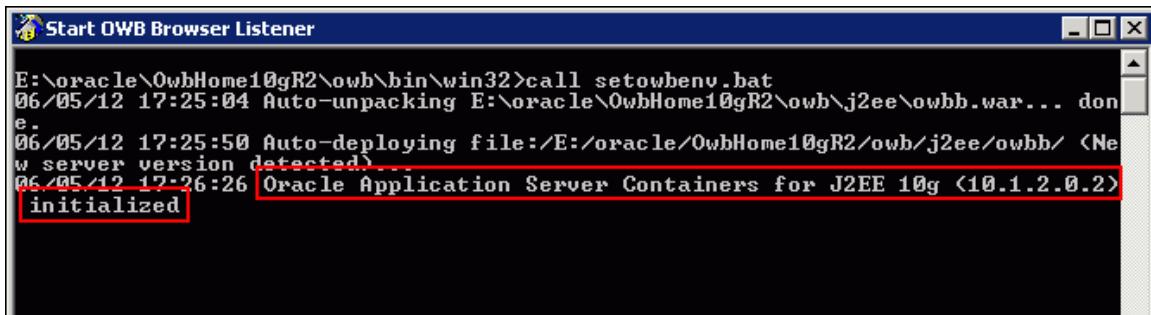
- 8) Start the OWB Browser Listener. Minimize the Start OWB Browser Listener window after you see that it has successfully started the browser listener.

Note: What you are working on in this classroom setup is a standalone version of the Repository Browser. In your working environment, you need to run one instance of the OWB Browser Listener in your enterprise. Then anyone with the login credentials and a Web browser can access OWB repository reports, without the need to have the OWB Client installed.

Answer:

1. Select Start > Programs > Oracle-OwbHome10gR2 > Warehouse Builder > Administration > Start OWB Browser Listener.
2. Wait till you see the message as shown in this screenshot:

Solutions for Practice 10-2: Metadata Management (continued)



The screenshot shows a Windows command-line window titled "Start OWB Browser Listener". The window displays deployment logs from the Oracle Warehouse Builder (OWB) application. The logs include:

```
E:\oracle\OwbHome10gR2\owb\bin\win32>call setowbenv.bat
06/05/12 17:25:04 Auto-unpacking E:\oracle\OwbHome10gR2\owb\j2ee\owbb.war... done.
06/05/12 17:25:50 Auto-deploying file:/E:/oracle/OwbHome10gR2/owb/j2ee/owbb/ (New server version detected)
06/05/12 17:26:26 Oracle Application Server Containers for J2EE 10g <10.1.2.0.2> initialized
```

3. Minimize this Browser Listener dialog box.

Start the Repository Browser

- 9) Launch the Warehouse Builder Repository Browser.

Answer:

Double-click the Repository Browser desktop icon to launch the Warehouse Builder Repository.

- 10) In the Browser window, enter the connection information details as:

Username: rep_owner

Password: rep_owner

Host address: localhost

Host Port Number: 1521

Host Service Name: orcl

Answer:

In the Browser window, enter the connection details as shown in the following screenshot, and click Login. The Control Center Browser window appears.

Solutions for Practice 10-2: Metadata Management (continued)

The screenshot shows the Oracle Warehouse Builder Browser Control Center. At the top, there is a navigation bar with links to Aria, Customize Links, Free Hotmail, My Oracle, Network Request, Oracle CRM, Oracle Email, and Help. The main title is "Warehouse Builder Browser". Below the title, there is a "Control Center" section. On the right side of this section are "Cancel", "Login", and "Help" buttons. The "Information" tab is selected, indicated by a green background. The text "Please supply connection details" is displayed. There are two sections for connection specification: one for "Specify the host address, port number, and service name" and one for "Specify the net service name". In the first section, the "UserName" field contains "rep_owner" and the "Password" field contains "*****". Under "Specify the host address, port number, and service name", the "Host Address" is "localhost", "Host Port Number" is "1521", and "Host Service Name" is "orcl". In the second section, the "Net Service Name" field is empty. At the bottom of the form, there is a copyright notice: "Copyright © 2000, 2006, Oracle. All rights reserved." and "Help".

Solutions for Practice 10-2: Metadata Management (continued)

ORACLE® Warehouse Builder Browser

Reports

Deployment

- [Deployment Schedule Report](#) - to display details of all deployments in time order
- [Object Summary Report](#) - to list all objects with details of their latest deployment
- [Locations Report](#) - to show all locations into which objects have been deployed

Execution

- [Execution Schedule Report](#) - to display details of all execution jobs in time order
- [Execution Summary Report](#) - to list all executable objects with details of their latest execution job

Management

- [Service Node Report](#) - to display and manage service node information for the RAC system

Related Links

 [Design Repository : Navigator](#)

Note: The Control Center of the Repository Browser provides the following three types of reports: Deployment Reports, Execution Reports, and Management Reports. Also available is the Design Objects reports.

Design Objects Reports

- 11) List the objects in the SALES_WH module. For security reasons, you will be prompted for the connection details. Provide the same details that you specified at the time of connection in step 10 of this practice.

Answer:

1. In the browser window, at the bottom of the page, in the Related Links section, click **Design Repository: Navigator**. The Navigator window appears.

Solutions for Practice 10-2: Metadata Management (continued)

The screenshot shows the Oracle Warehouse Builder Browser interface. At the top, the title "Warehouse Builder Browser" is displayed next to the Oracle logo. Below the title, there is a "Navigator" section with a "Search" bar containing fields for "Search By Type" and "Name" with a "Go" button. The main area is titled "All" and contains a tree view of metadata objects. A "Refresh" button is located above the tree. The tree structure shows two main nodes: "Top" (indicated by a downward arrow) and "rep_owner" (indicated by a right-pointing arrow). The "rep_owner" node is expanded, revealing its type as "Repository". To the right of the tree, there is a table with columns: Focus, Name, Type, Reports, Lineage, and Impact. The "Top" node has a lineage value of 60 and an impact value of 60. The "rep_owner" node also has a lineage value of 60 and an impact value of 60. Below the tree, there is a "Related Links" section with a link to "Control Center : Reports".

Focus	Name	Type	Reports	Lineage	Impact
	Top			60	60
rep_owner	rep_owner	Repository		60	60

2. In the Navigator, expand rep_owner > CLASS_PROJECT > SALES_WH. For security reasons, you will be prompted to enter the connection details. Enter the connection details as in step 10.

Solutions for Practice 10-2: Metadata Management (continued)

Focus	Name	Type	Reports	Lineage	Impact
⊕	▼ 🏭 SALES_WH	Oracle Module	⊕	60	60
⊕	▶ 📈 # SALES	Cube	⊕	60	60
⊕	▶ ↗ CHANNELS	Dimension	⊕	60	60
⊕	▶ ↗ CUSTOMERS	Dimension	⊕	60	60
⊕	▶ ↗ PRODUCTS	Dimension	⊕	60	60
⊕	▶ ↗ PROMOTIONS	Dimension	⊕	60	60
⊕	▶ ↗ TIMES	Dimension	⊕	60	60
⊕	▶ 123 CLASS_DIM_SEQ	Sequence	⊕	60	60
⊕	▶ 123 CUST_DIM_SEQ	Sequence	⊕	60	60
⊕	▶ 123 PROD_DIM_SEQ	Sequence	⊕	60	60
⊕	▶ 123 PROMO_DIM_SEQ	Sequence	⊕	60	60
⊕	▶ 123 TIME_DIM_SEQ	Sequence	⊕	60	60
⊕	▶ 📈 CHANNELS	Table	⊕	60	60
⊕	▶ 📈 CUSTOMERS	Table	⊕	60	60

Browse Control Center Reports

- 12) Open the Control Center Reports page.

Answer:

To browse the Control Center Reports, at the bottom of the page, in the Related Links section, click Control Center: Reports. The reports are categorized into Deployment, Execution, and Management reports.

Solutions for Practice 10-2: Metadata Management (continued)

Navigator

Search

Search By Type

All

Focus Name Type Reports Lineage Impact

Focus	Name	Type	Reports	Lineage	Impact
▼	Top				
⊕	rep_owner	Repository			

Related Links

[Control Center : Reports](#)

Copyright © 2000, 2006, Oracle. All rights reserved.

[Logout](#) | [Help](#)

Object Summary Report

13) Examine the object summary report of all the objects.

Answer:

In the Deployment section, click **Object Summary Report**.

Reports

Deployment

- [Deployment Schedule Report](#) - to display details of all deployments in time order
- [Object Summary Report](#) - to list all objects with details of their latest deployment
- [Locations Report](#) - to show all locations into which objects have been deployed

Execution

- [Execution Schedule Report](#) - to display details of all execution jobs in time order
- [Execution Summary Report](#) - to list all executable objects with details of their latest execution job

Management

- [Service Node Report](#) - to display and manage service node information for the RAC system

Related Links

[Design Repository : Navigator](#)

Solutions for Practice 10-2: Metadata Management (continued)

Reports : rep_owner >

Object Summary Report

Repository rep_owner	Type RUNTIME REPOSITORY	Installed Version 10.2.0.1.31	Available Reports
Description			Deployment Schedule
			Object Summary
			Locations
			Execution Schedule
			Execution Summary

Deployments

Filter on Object Type

Filter on Object Status

Name	Type	Location	Latest Deployment	Obj Status
SALES	Cube	SALES_WH_LOCATION	2006-05-18 16:02:00	Valid
CHANNELS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
CUSTOMERS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
PRODUCTS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
PROMOTIONS	Dimension	SALES_WH_LOCATION	2006-05-18 15:10:20	Valid
TIMES	Dimension	SALES_WH_LOCATION	2006-05-18 16:01:40	Valid
LOAD_CHANNELS	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 15:39:52	Valid
LOAD_CUSTOMERS	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 16:03:20	Valid

14) Examine the execution status of the objects.

Answer:

To examine the execution status, on the Object Summary Report page, on the right of the Available Reports page, select Execution Summary.

Execution Jobs

Filter on Type

Filter on Execution Status

Name	Type	Location	Latest Execution	Execution Status	Related Information
LOAD_CHANNELS	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 15:40:05	Complete : OK	Execution Job Report
TIMES_MAP	PL/SQL Map	SALES_WH_LOCATION	2006-05-18 15:40:50	Complete : OK	Execution Job Report

15) Examine the complete details about the execution of the LOAD_CHANNELS mapping.

Answer:

To get complete details about the execution of the LOAD_CHANNELS mapping, click Execution Job Report in the Related Information column of the table.

16) Log out from the repository browser.

Answer:

- At the bottom of the repository browser window, click Logout to log out from the Warehouse Builder Repository.
- Close the Browser window.

17) Stop the OWB Browser Listener.

Answer:

18) Click Start > Programs > Oracle-OwbHome10gR2 > Warehouse Builder > Administration > Stop OWB Browser Listener.

Practice Solutions for Lesson 11

Your CLASS_PROJECT project contains a target module SALES_WH that implements the relational data warehouse. The first set of exercises applies to business definitions, enabling the seamless integration with Oracle Discoverer. The second part enables you to perform ad hoc queries on top of relational structures.

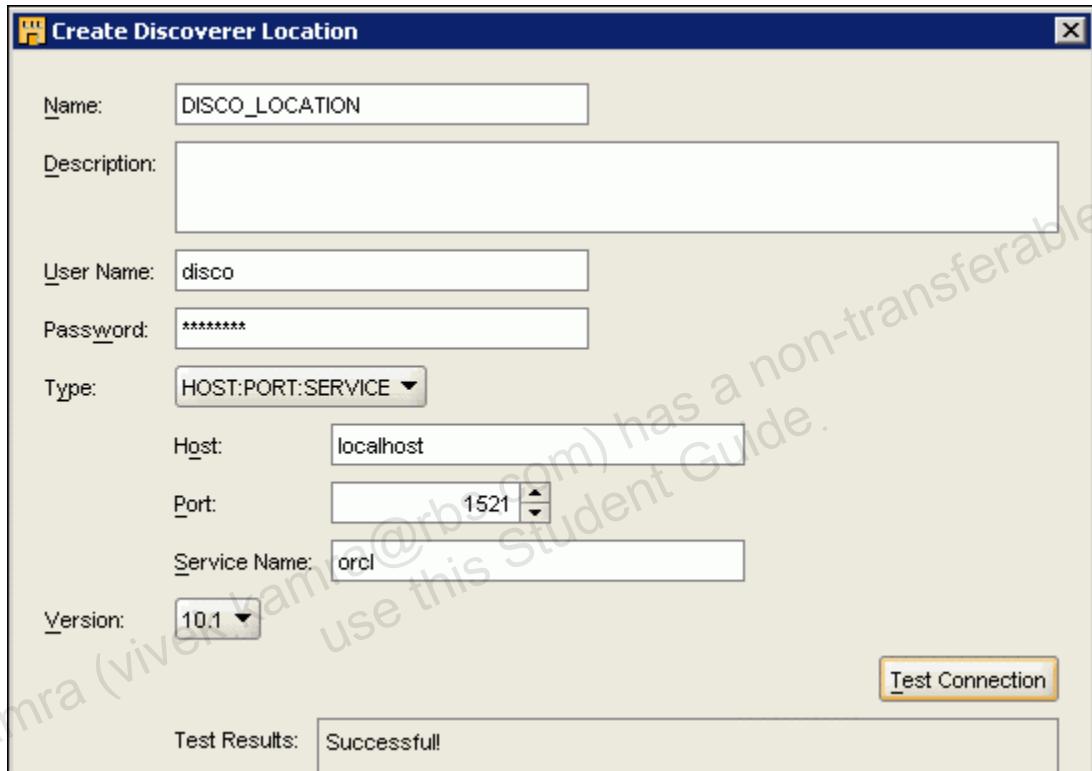
vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects

Define a Discoverer Location

A Discoverer location provides details about the system to which the business definitions you create are deployed. This system should have an Oracle Discoverer EUL version 10.1.2 or later installed.

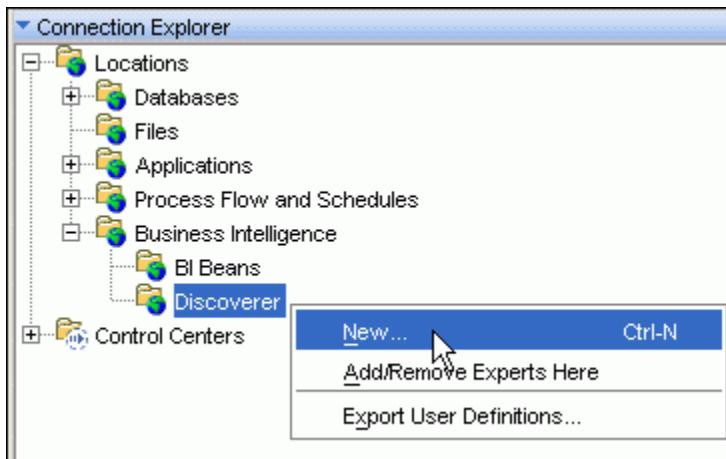
- 1) Define a Discoverer location with the following details:



Answer:

1. In the Design Center, in the Connection Explorer, expand **Business Intelligence**. Right-click **Discoverer**, and select **New**. The Create Discoverer Location dialog box appears.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



2. In the Create Discoverer Location dialog box, enter the following details:

Name: **DISCO_LOCATION**

User Name: **disco**

Password: **disco**

Host: **localhost**

Port: **1521**

Service Name: **orcl**

Define a Business Definition Module

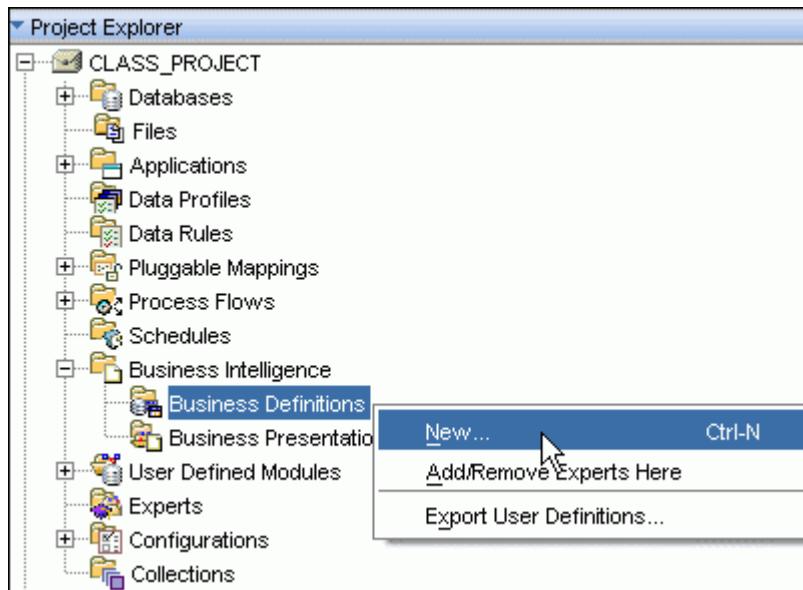
In the previous exercises, you created different kinds of modules—for example, Oracle modules, files module, process flow modules. Modules are storage objects that help you organize objects within a module. Now you create a module to store the business definitions.

- 2) Create a business definition module named DISCOVERER_OBJECTS.

Answer:

1. In the Project Explorer, expand **Business Intelligence**. Right-click **Business Definitions**, and select **New**. Click **Next** on the Welcome page.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

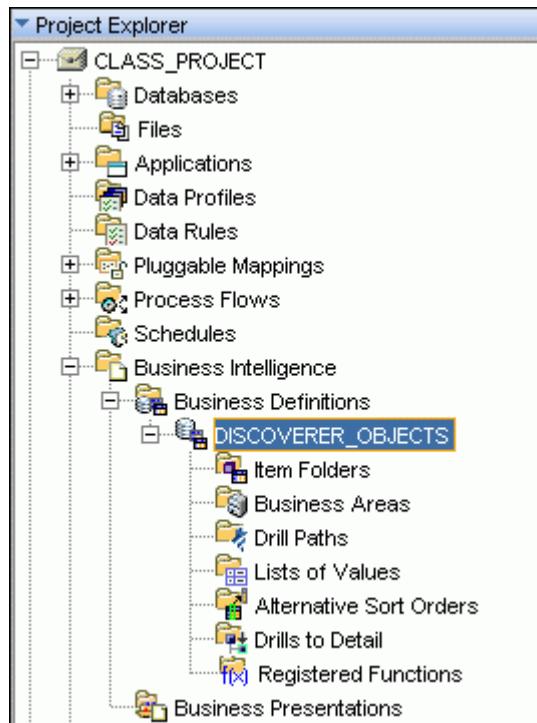


2. In the Name field on the Name and Description page, enter **DISCOVERER_OBJECTS**, and click **Next**.
 3. On the Connection Information page, from the Location drop-down list, select **DISCO_LOCATION** (the location you just created). This location will be the deployment target for the objects you create. Review the details, and click **Next**.
 4. On the Summary page, review the summary information, and click **Finish**.
- 3) Locate and examine DISCOVERER_OBJECTS, the Business Definition module that you just created in the Design Center.

Answer:

In the Project Explorer, expand the DISCOVERER_OBJECTS module. Notice that Warehouse Builder provides a representation for most objects you find in a Discoverer End User Layer (EUL). Warehouse Builder uses the terminology that Discoverer exposes.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



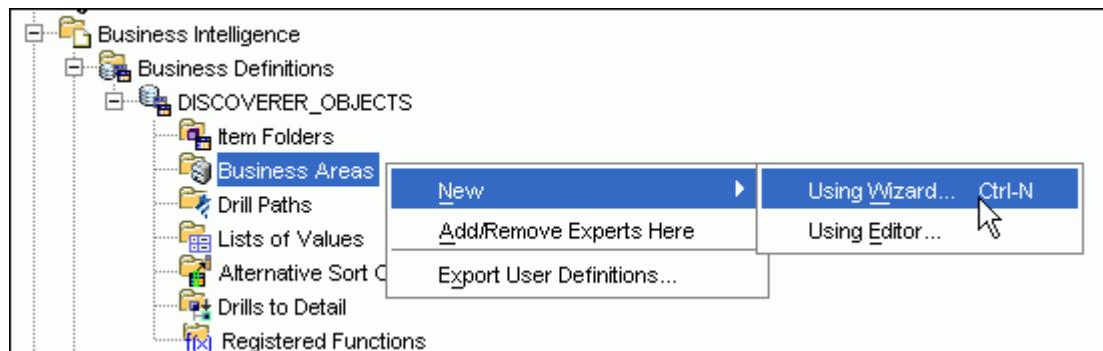
Define a Business Area

Business areas contain references to Item Folders stored in your Business Definition module. Warehouse Builder enables you to create a business area to deploy to a Discoverer EUL.

- 4) Create a business area named SALES_DATA.

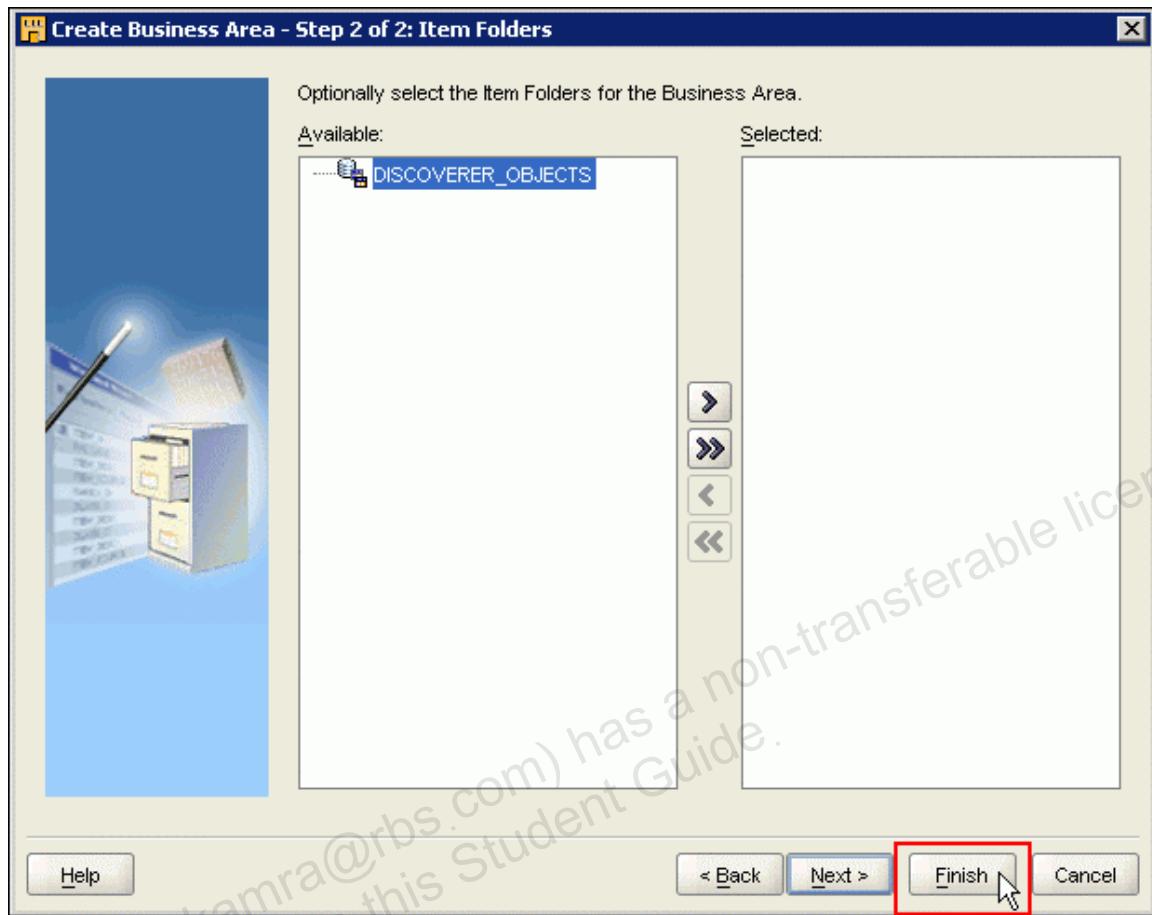
Answer:

1. In the Project Explorer, right-click **Business Areas**, and select **New > Using Wizard**. Click **Next** on the Welcome page.



2. On the Name and Description page, enter the name **SALES_DATA**, and click **Next**.
3. On the wizard's Item Folders page, accept the default selection, and click **Finish**. You do not see any item folders at this time because you will create one in the next section.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



Create a Simple Item Folder

First, you create a simple item folder. Later, you create a complex item folder.

- 5) Create a Simple Item Folder named SALES_PROMOTIONS using the editor. The SALES_PROMOTION item folder is based on the **SALES_WH.PROMOTIONS** table. Connect the columns from the PROMOTIONS table to the item folder.

Note: There are two nodes available for defining item folders. One is at the lower Business Areas level (SALES_DATA). The other is at the higher Business Definition Module level (DISCOVERER_OBJECTS). For this example, you create an item folder at the higher module level.

Answer:

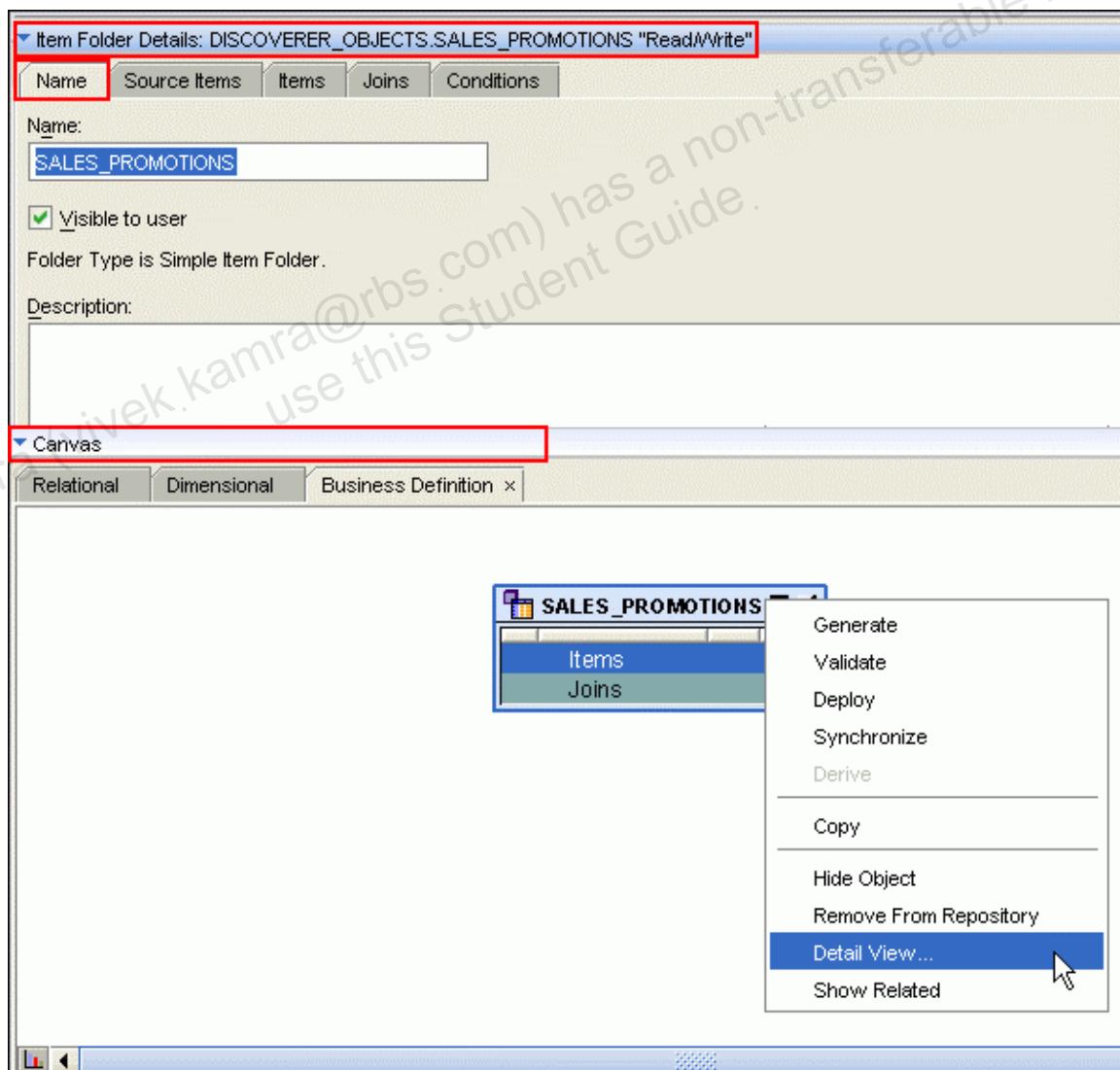
1. Right-click the **Item Folders** node located immediately under the DISCOVERER_OBJECTS module (not the node under the SALES_DATA business area), and select **New > Using Editor**.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



2. The Data Object Editor opens with an item folder having the default name ITEM_FOLDER_1. In the Item Folder Details panel, ensure that the Name tabbed page is displayed. In the Name field, change the item folder name to SALES_PROMOTIONS.

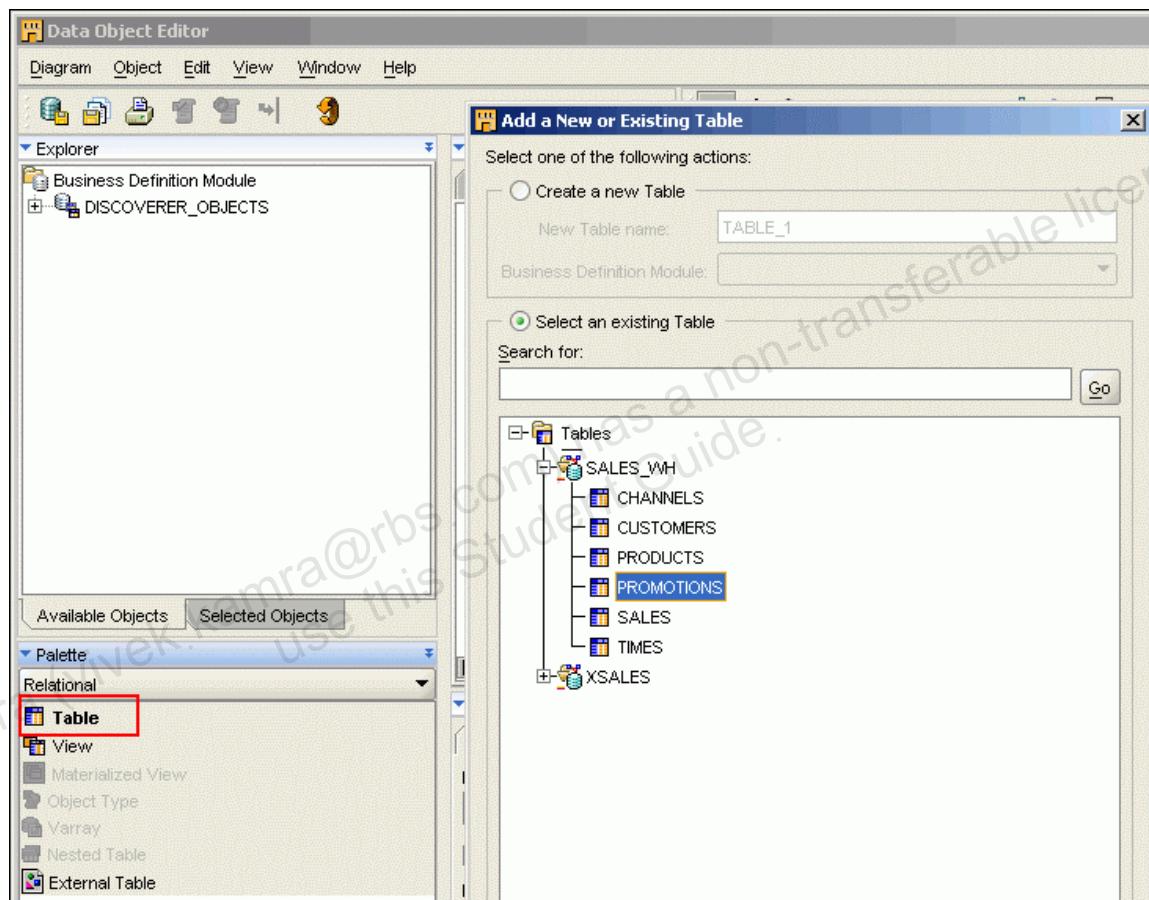
Note: If the Item Folder Details panel is not visible, select **Object Details** from the **Window** menu.



Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

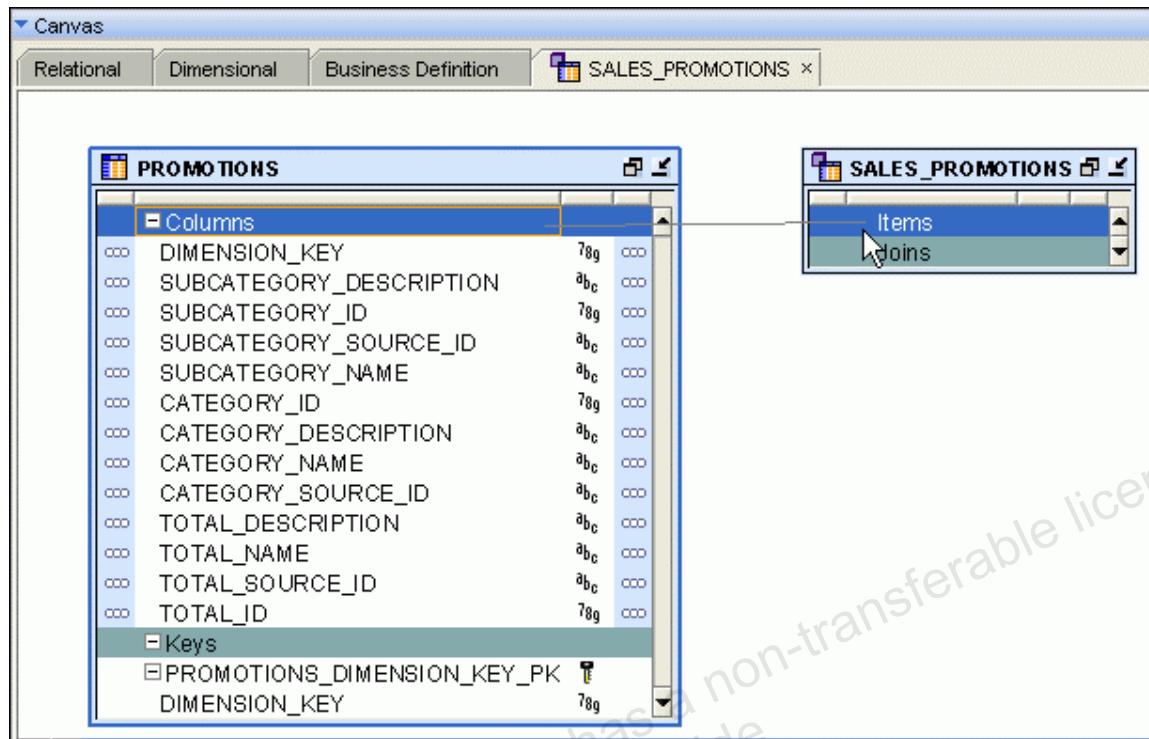
3. On the Canvas, right-click the header area of SALES_PROMOTION, and select **Detail View**.
4. Drag the Table icon from the Palette to the canvas. The Add a New or Existing Table dialog box appears. Expand **Tables > SALES_WH**, and select the **PROMOTIONS** table. Click **OK**.

Note: If you are not able to see the PROMOTIONS object anywhere on the canvas, click the Auto Layout icon on the toolbar.

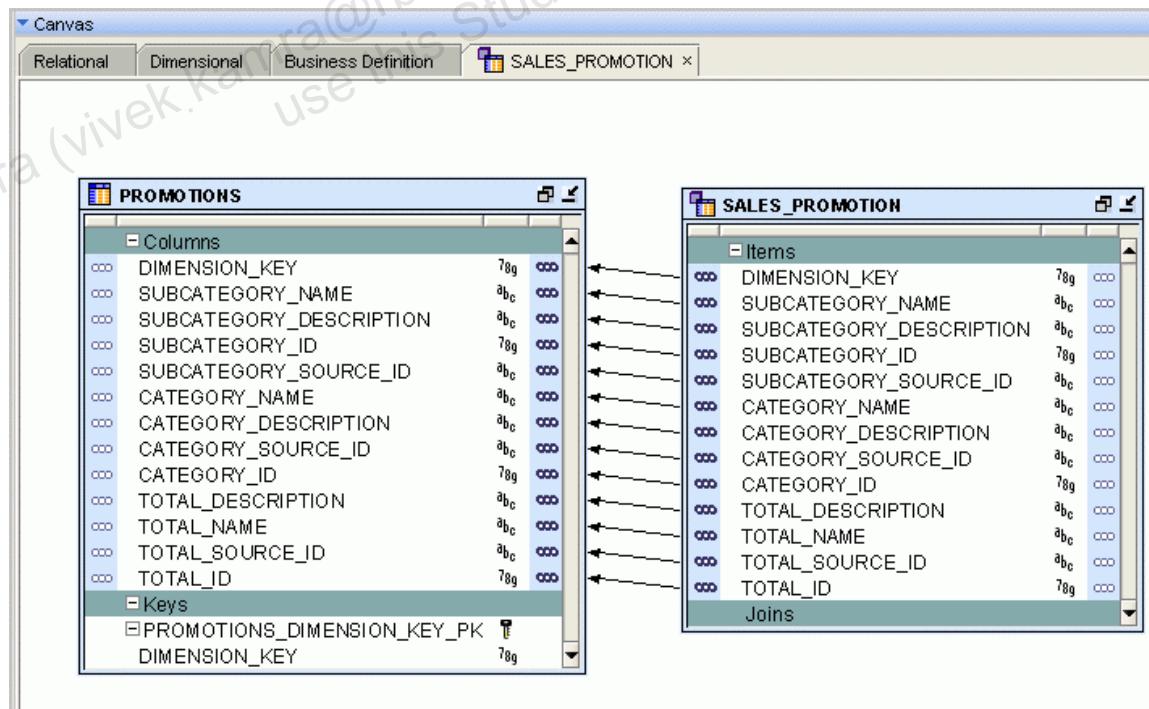


5. Place the PROMOTIONS table to the left of the SALES_PROMOTIONS item folder.
6. From the Columns header of the PROMOTIONS table, drag a line to the Items node in the SALES_PROMOTIONS item folder.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



7. This brings all of the columns from the PROMOTIONS table operator to the SALES_PROMOTIONS item folder.



- 6) Save your work, and close the editor.

Answer:

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

In the Data Object Editor, click **Save All** on the toolbar. From the menu, select **Diagram > Close Window**.

Create a Drill Path

A drill path primarily defines a hierarchy relationship in a dimension. Warehouse Builder enables you to create a drill path to deploy to a Discoverer EUL.

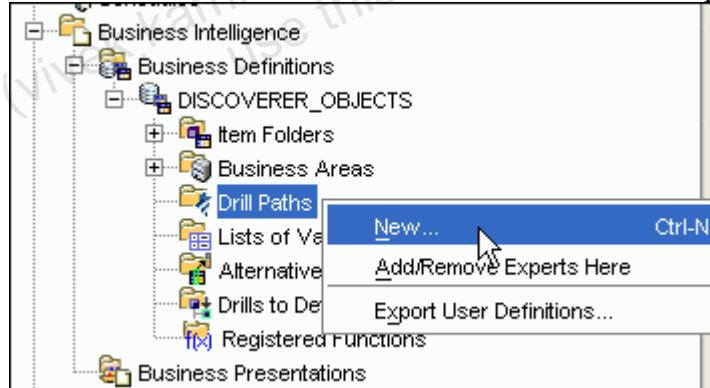
Besides creating drill paths manually, you can use a wizard to create Drills to Detail. Alternately, if you derive a dimension, you get the drill path automatically.

- 7) Create a new drill path, PROMOTIONS_DRILL, under the DISCOVERER_OBJECTS business definition module. Create a drill path, with drill levels in the following order:

Drill Level	Item Folder	Drill Level Item
TOTAL	SALES_PROMOTIONS	TOTAL_NAME
SUB_CATEGORY	SALES_PROMOTIONS	SUBCATEGORY_NAME
CATEGORY	SALES_PROMOTIONS	CATEGORY_NAME

Answer:

1. In the Project Explorer, right-click **Drill Paths**, and select **New**. The Create Drill Path Wizard is launched. Click **Next** on the Welcome page.



2. On the Name and Description page, enter **PROMOTIONS_DRILL** as the name and click **Next**.
3. On the Drill Levels page, specify the following details. (Create the drill levels in the order shown in the table.)

Drill Level	Item Folder	Drill Level Item
TOTAL	SALES_PROMOTIONS	TOTAL_NAME
SUB_CATEGORY	SALES_PROMOTIONS	SUBCATEGORY_NAME
CATEGORY	SALES_PROMOTIONS	CATEGORY_NAME

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Create Drill Path - Step 2 of 2: Drill Levels

Drill Levels:

	Drill Level	Item Folder	Description
1	TOTAL	SALES_PROMOTIONS	

Drill Level Items:

	Item	Use as drill item
1	DIMENSION_KEY	<input type="checkbox"/>
2	SUBCATEGORY_DESC...	<input type="checkbox"/>
3	SUBCATEGORY_ID	<input type="checkbox"/>
4	SUBCATEGORY_SOUR...	<input type="checkbox"/>
5	SUBCATEGORY_NAME	<input type="checkbox"/>
6	CATEGORY_ID	<input type="checkbox"/>
7	CATEGORY_DESCRIPTI...	<input type="checkbox"/>
8	CATEGORY_NAME	<input type="checkbox"/>
9	CATEGORY_SOURCE_ID	<input type="checkbox"/>
10	TOTAL_DESCRIPTION	<input type="checkbox"/>
11	TOTAL_NAME	<input checked="" type="checkbox"/>
12	TOTAL_SOURCE_ID	<input type="checkbox"/>
13	TOTAL_ID	<input type="checkbox"/>

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Create Drill Path - Step 2 of 2: Drill Levels

Drill Levels:

	Drill Level	Item Folder	Description
1	TOTAL	SALES_PROMOTIONS	
2	SUB_CATEGORY	SALES_PROMOTIONS	

Drill Level Items:

	Item	Use as drill item
1	DIMENSION_KEY	<input type="checkbox"/>
2	SUBCATEGORY_DESC...	<input type="checkbox"/>
3	SUBCATEGORY_ID	<input type="checkbox"/>
4	SUBCATEGORY_SOUR...	<input type="checkbox"/>
5	SUBCATEGORY_NAME	<input checked="" type="checkbox"/>
6	CATEGORY_ID	<input type="checkbox"/>
7	CATEGORY_DESCRIPTI...	<input type="checkbox"/>
8	CATEGORY_NAME	<input type="checkbox"/>
9	CATEGORY_SOURCE_ID	<input type="checkbox"/>
10	TOTAL_DESCRIPTION	<input type="checkbox"/>
11	TOTAL_NAME	<input type="checkbox"/>
12	TOTAL_SOURCE_ID	<input type="checkbox"/>
13	TOTAL_ID	<input type="checkbox"/>

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

The screenshot shows the 'Create Drill Path - Step 2 of 2: Drill Levels' dialog box. It contains two main sections: 'Drill Levels:' and 'Drill Level Items:'.

Drill Levels:

	Drill Level	Item Folder	Description
1	TOTAL	SALES_PROMOTIONS	
2	SUB_CATEGORY	SALES_PROMOTIONS	
3	CATEGORY	SALES_PROMOTIONS	

Drill Level Items:

	Item	Use as drill item
1	DIMENSION_KEY	<input type="checkbox"/>
2	SUBCATEGORY_DESCRIPTION	<input type="checkbox"/>
3	SUBCATEGORY_ID	<input type="checkbox"/>
4	SUBCATEGORY_SOURCE_ID	<input type="checkbox"/>
5	SUBCATEGORY_NAME	<input type="checkbox"/>
6	CATEGORY_ID	<input type="checkbox"/>
7	CATEGORY_DESCRIPTION	<input type="checkbox"/>
8	CATEGORY_NAME	<input checked="" type="checkbox"/>
9	CATEGORY_SOURCE_ID	<input type="checkbox"/>
10	TOTAL_DESCRIPTION	<input type="checkbox"/>
11	TOTAL_NAME	<input type="checkbox"/>
12	TOTAL_SOURCE_ID	<input type="checkbox"/>
13	TOTAL_ID	<input type="checkbox"/>

4. Click Next. Review the summary information, and click Finish to close the Create Drill Path Wizard.
- 8) Save your work.

Answer:

In the Design Center, click the Save All icon on the toolbar.

Derive Individual Objects

You have now created a number of structures for Discoverer manually. In previous lessons, you defined a large amount of metadata as data object definitions. Rather than manually adding more item folders and drill paths, you can reuse your metadata definitions and propagate them into business definitions.

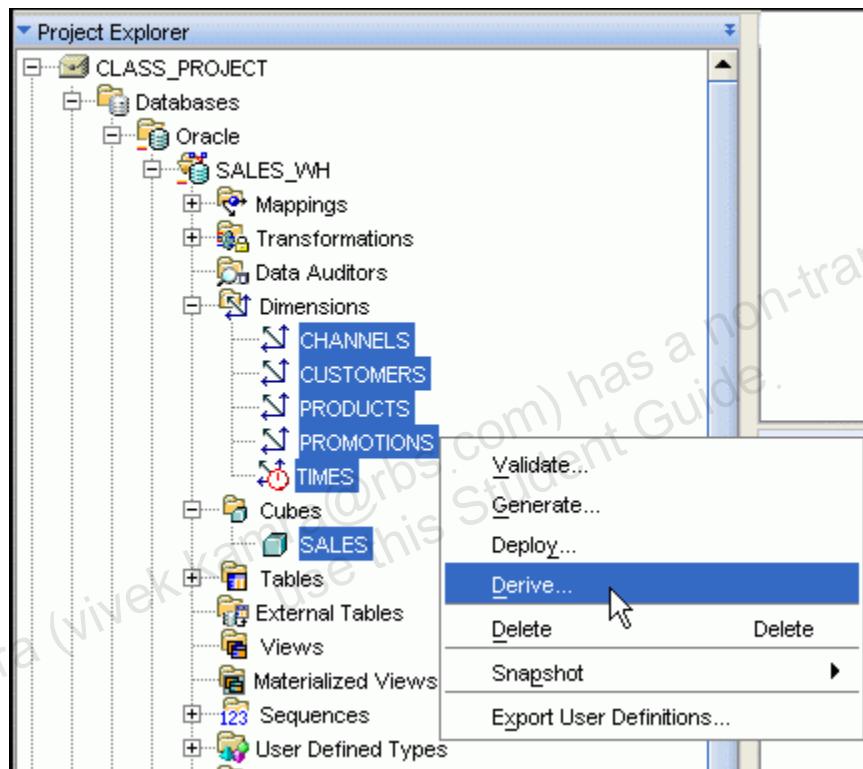
With Oracle Warehouse Builder 10g Release 2, you can derive sets of individual data objects or a grouping of data objects in the form of a collection. The following steps guide you through both scenarios.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

- 9) Derive the dimensions and the cube from the SALES_WH target module using the Perform Derivation Wizard.

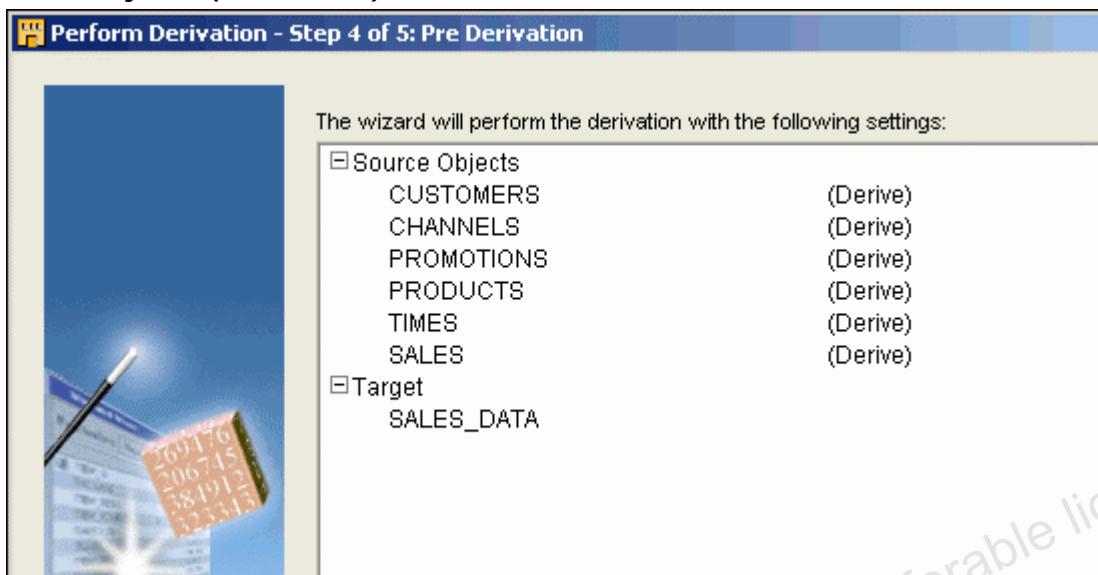
Answer:

1. In the Project Explorer, expand **SALES_WH >Dimensions**. Expand **Cubes**. Use the **Ctrl** key to multiselect all dimensions and the cube.
2. With the five dimensions and one cube selected, press and hold the **Ctrl** key, right-click the selected objects, and select **Derive**. The Perform Derivation Wizard is launched. Click **Next** on the Welcome page.

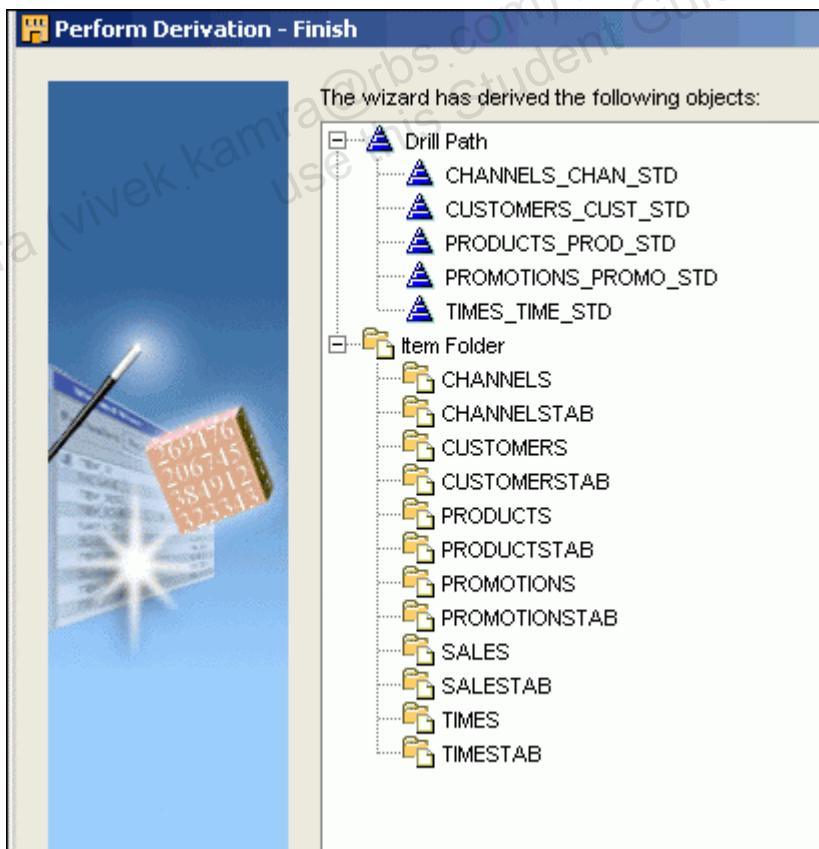


3. On the Source Objects page, ensure that the five dimensions and one cube you selected appear in the Selected list on the right. Click **Next**.
4. On the wizard's Target page, select the **SALES_DATA** business area as the target. Click **Next**.
5. Review the rules on the Rules page, without changing any. The rules influence the behavior of the derivation process. Click **Next**.
6. On the wizard's Pre Derivation page, confirm that all five dimensions and the cube are listed. Click **Next**.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



7. Wait for the derivation to complete, and verify that the message log shows "Information: Derivation finished."
8. Click Next. On the wizard's Finish page, examine all the drill paths and item folders that were derived. Click Finish.



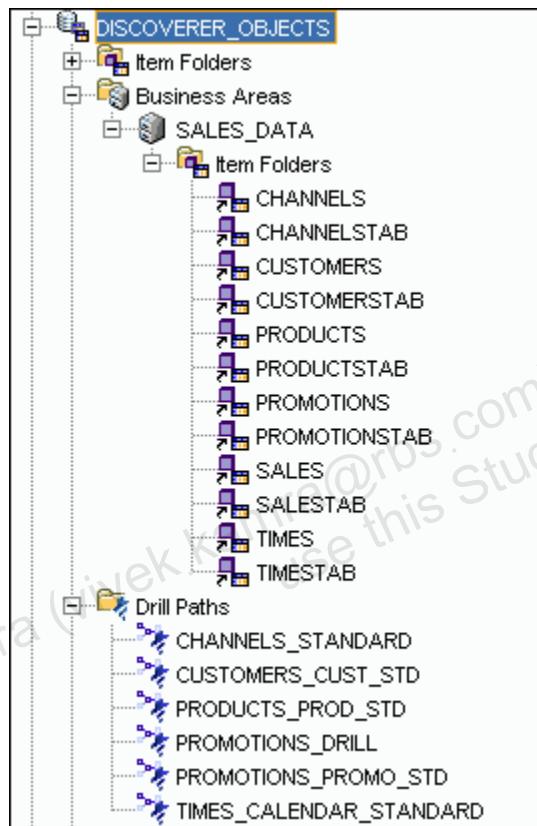
- 10) Open the SALES_DATA business area. Verify that the tree shows the definitions that were derived.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Warehouse Builder creates hidden Simple Item Folders on top of the relational tables that implement the dimensions and the cube, and it creates Complex Item Folders on top of the Simple Item Folders, which are visible to users. The derivation process also creates the drill paths that implement the respective hierarchies on the dimensions. Finally, because you selected a business area as a target, all item folders are referenced by that business area.

Answer:

1. In the Project Explorer, expand DISCOVERER_OBJECTS > Business Areas > SALES_DATA > Item Folders. Expand Drill Paths.



Derive Objects from a Collection

You just derived business objects from five dimensions and a cube. In addition to deriving from individual object definitions, you can derive from OWB “collections.” You create a collection, and then derive business objects from that collection.

Create a New Business Definitions Module

- 11) Create a new Business Definitions module named MART_DEFINITIONS. On the wizards Connection Information page, use DISCO_LOCATION.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



Answer:

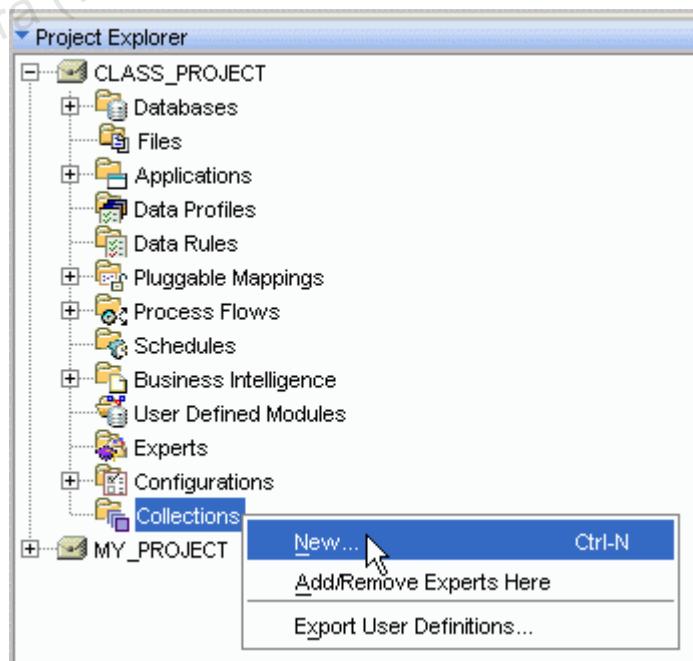
1. Under the Business Intelligence > Business Definitions node, collapse the DISCOVERER_OBJECTS node.
2. Right-click **Business Definitions**, and select **New**. The Create Business Definition Module Wizard is launched. Click **Next** on the Welcome page.
3. Create a new Business Definitions module named **MART_DEFINITIONS**. On the wizard's Connection Information page, select DISCO_LOCATION from the Location drop-down list. Review the connection details, and click **Next**.
4. Review the summary information, and click **Finish** to close the wizard.

Create a Collection

- 12) Create a collection named SALES_MART. The collection refers to two dimensions and one cube, namely, the CHANNELS and PRODUCTS dimensions and the SALES cube.

Answer:

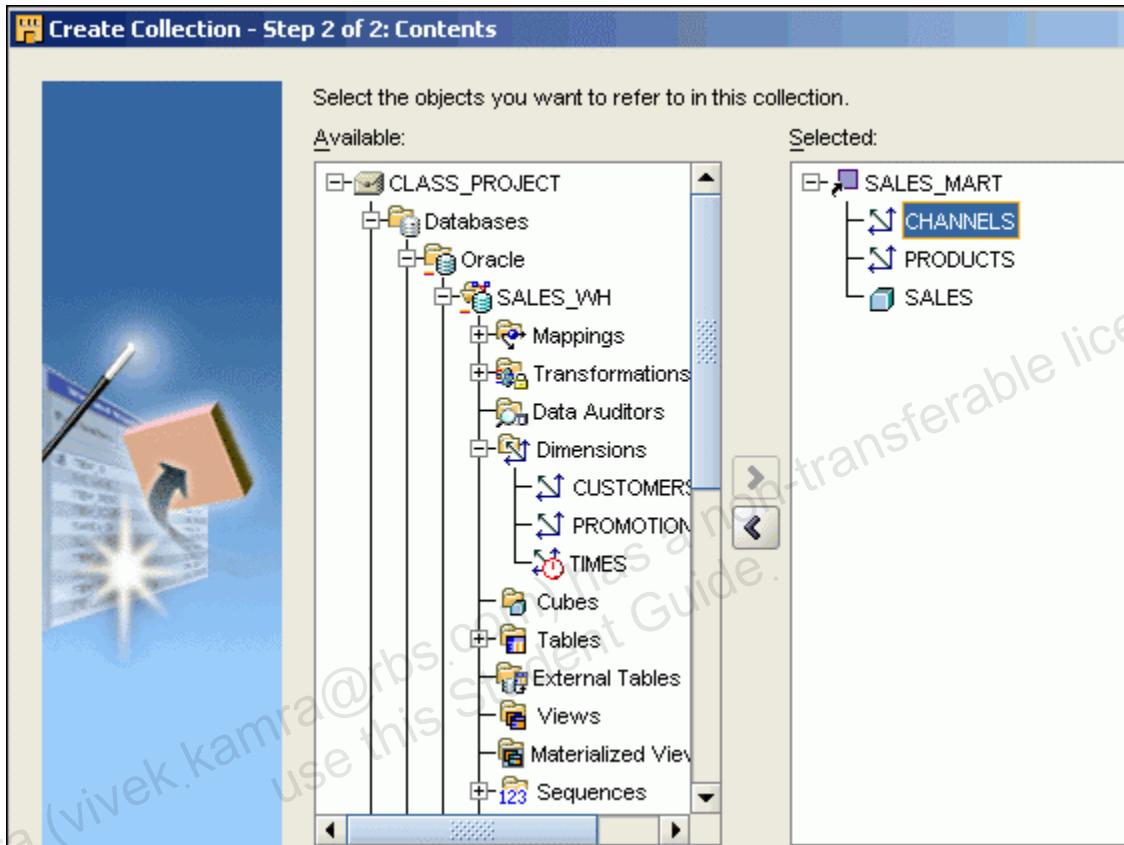
1. In the Project Explorer, right-click **Connections**, and select **New**. Click **Next** on the Welcome page.



2. On the Name and Description page, enter **SALES_MART**.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

3. On the Contents page, expand **CLASS_PROJECT > Databases > Oracle > SALES_WH > Dimensions**. Expand Cubes. Press and hold the Ctrl key, and select **CHANNELS** and **PRODUCTS** dimensions and the **SALES** cube, and move them to the Selected list. Click **Next**. Click **Finish**.



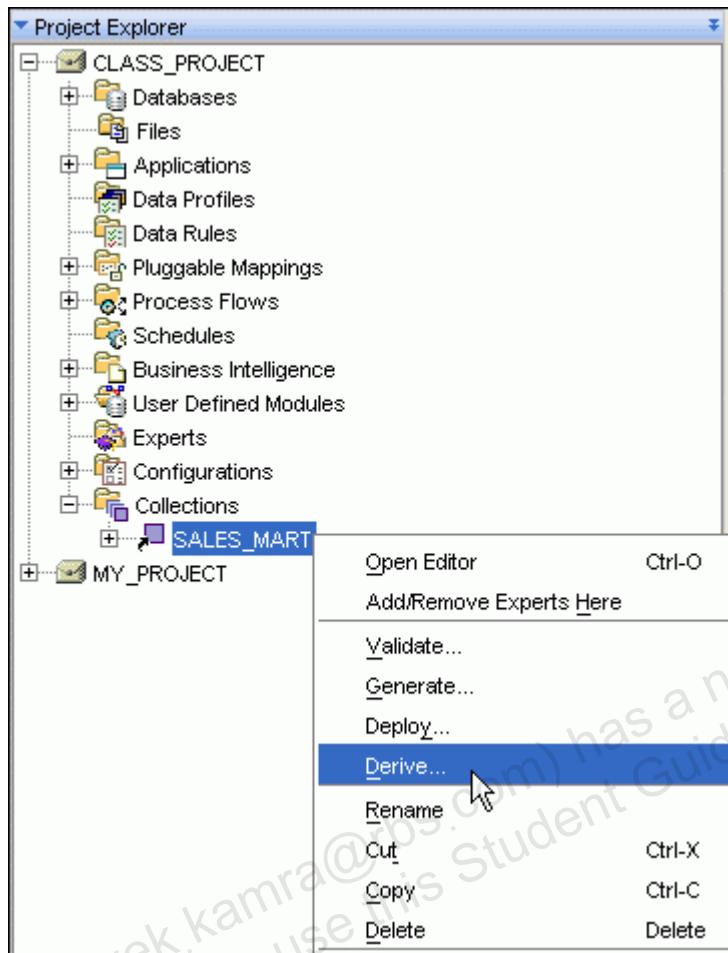
Derive Business Definitions from the SALES_MART Collection

- 13) Derive business definitions from the **SALES_MART** collection. Select the **MART_DEFINITIONS** business definition module to contain the derived objects.

Answer:

1. Right-click the **SALES_MART** collection, and select **Derive**. The Perform Derivation Wizard is launched.

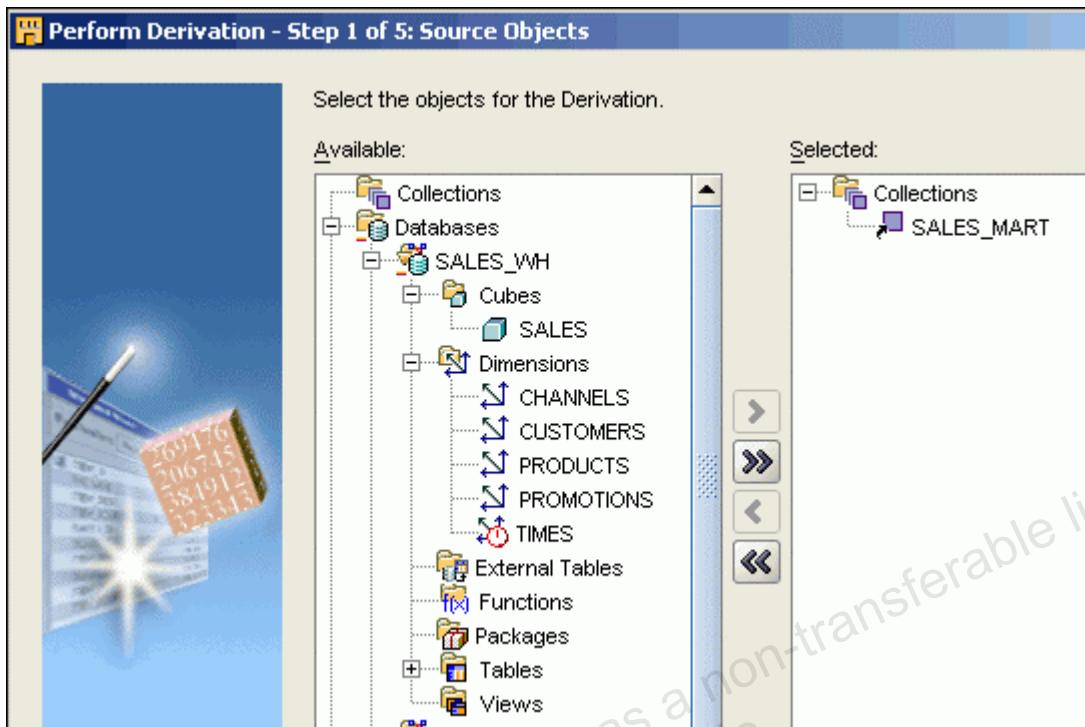
Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



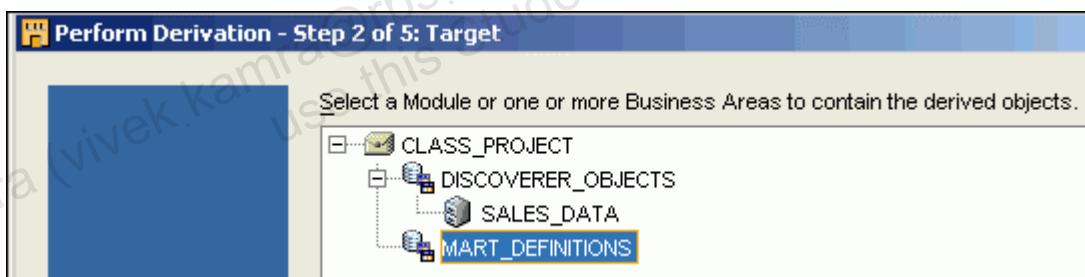
2. On the wizard's Source Objects page, select SALES_MART, and shuttle it to the Selected list.

Note: In the Selected list, if you see any other objects, other than the SALES_MART object, shuttle them back to the Available list by using the << icon.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

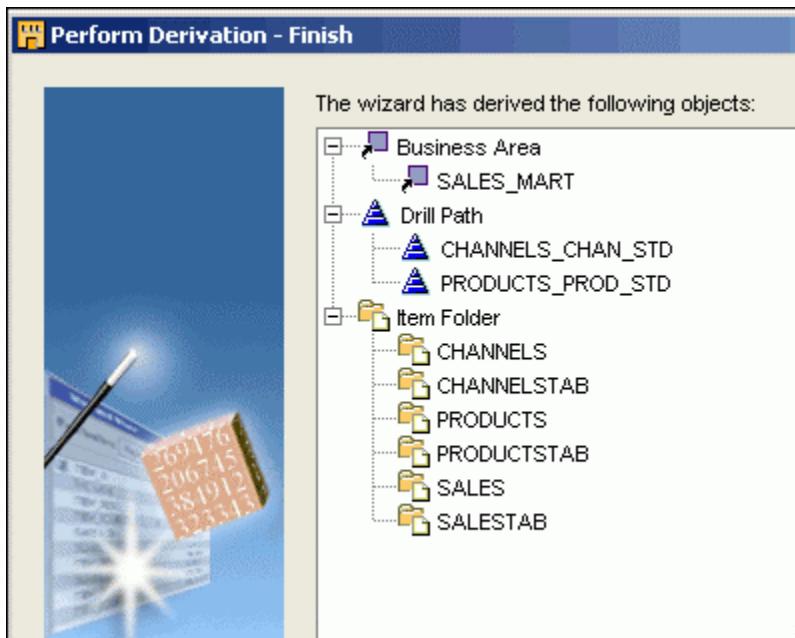


3. Click Next. The wizard's Target page appears. Select **MART_DEFINITIONS**, the business definition module you created earlier in this practice. Click Next.



4. Review the rules, without changing any. Click Next.
5. On the Pre Derivation page, click Next. Wait for the derivation finished message, and then click Next.
6. Notice that Warehouse Builder created a business area, drill paths for the dimension hierarchies, and item folders for the dimensions and cube. Click Finish.

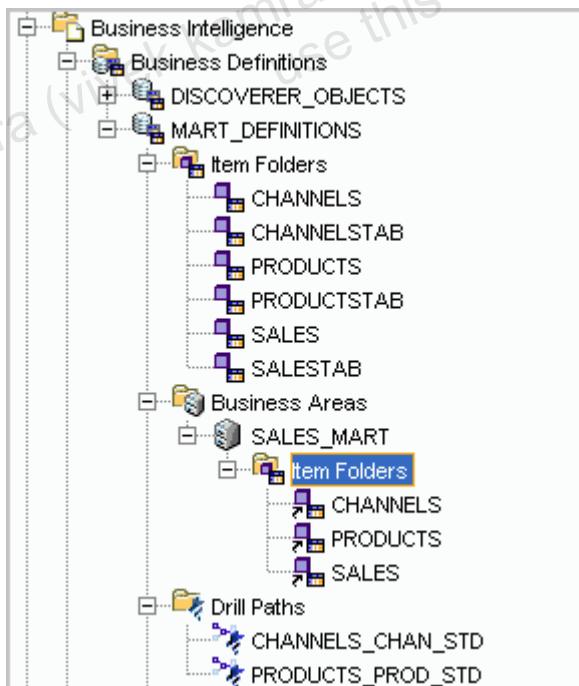
Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



- 14) Examine the objects created in SALES_MART.

Answer:

In the Project Explorer, expand **Business Definitions**. Select **MART DEFINITIONS** from the View menu, and select Expand All. Note the item folders derived from the SALES_MART collection you created.



By allowing you to derive from a collection, Warehouse Builder enables you to derive business definitions from a group of objects that you organized into a SALES_MART collection. Then, in one step, you derived all the required objects

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

based on your own grouping, without having to search through all the objects to understand which ones were required.

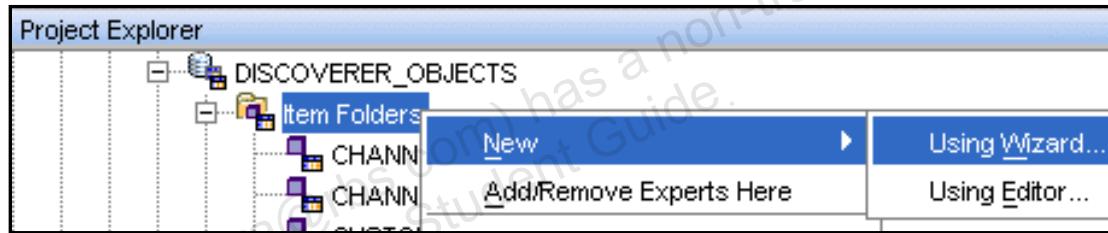
Graphically Create a Complex Item Folder

Your manager always wants to see the sales information in a tabular format, categorized by product. The objective is to see the actual sales and costs, relative to the actual list prices. To create a report that represents these details as easily as possible, you decide to create a complex folder to represent this data.

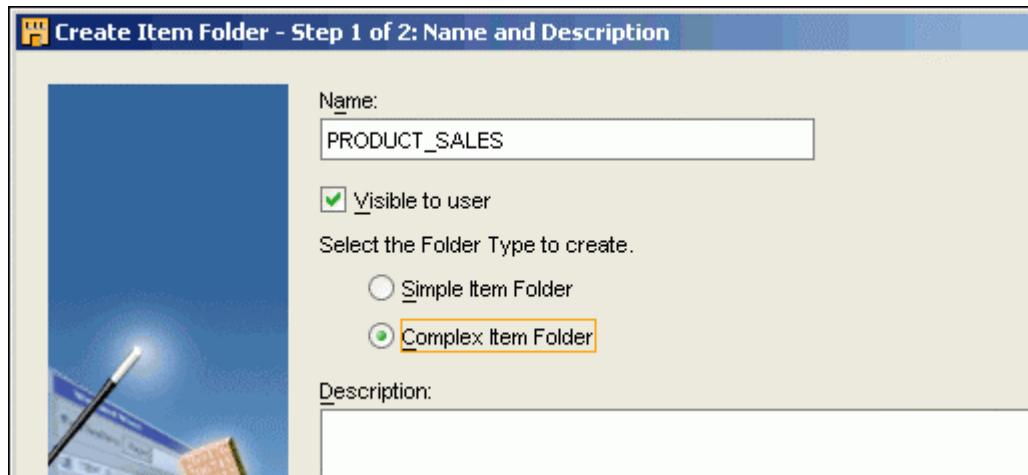
- 15) Create a complex folder named PRODUCT_SALES. Select PRODUCTS and SALES as the source items

Answer:

1. In the Project Explorer, expand Business Intelligence > Business Definitions > DISCOVERER_OBJECTS.
2. Right-click the Item Folders node immediately under DISCOVERER_OBJECTS, and select New > Using Wizard.

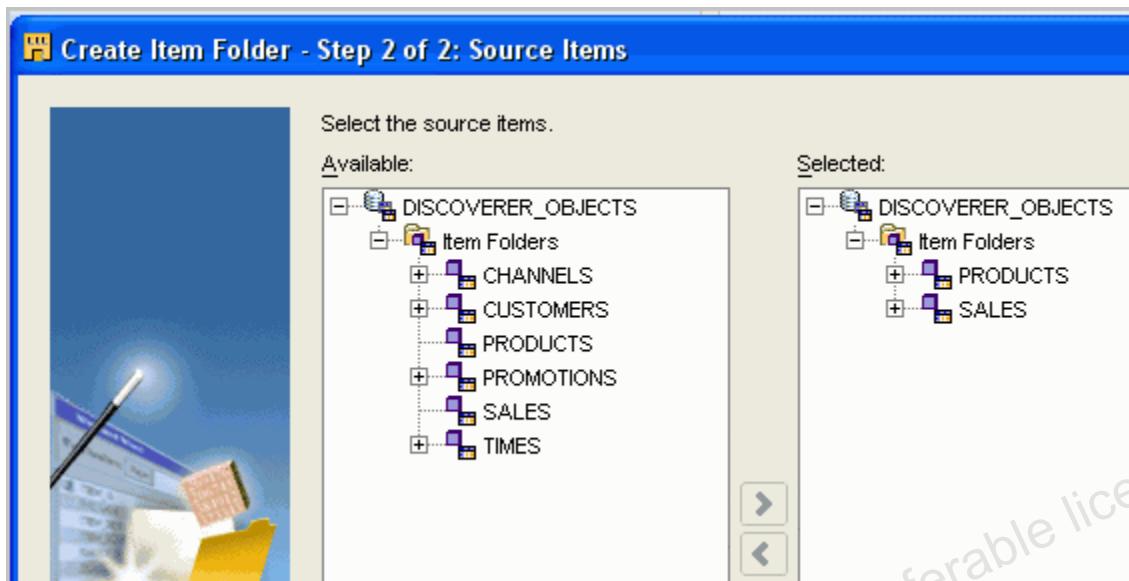


3. On the wizard's Name and Description page, name this complex folder as PRODUCT_SALES, keep the "Visible to user" check box selected, and select Complex Item Folder as the folder type. Click Next

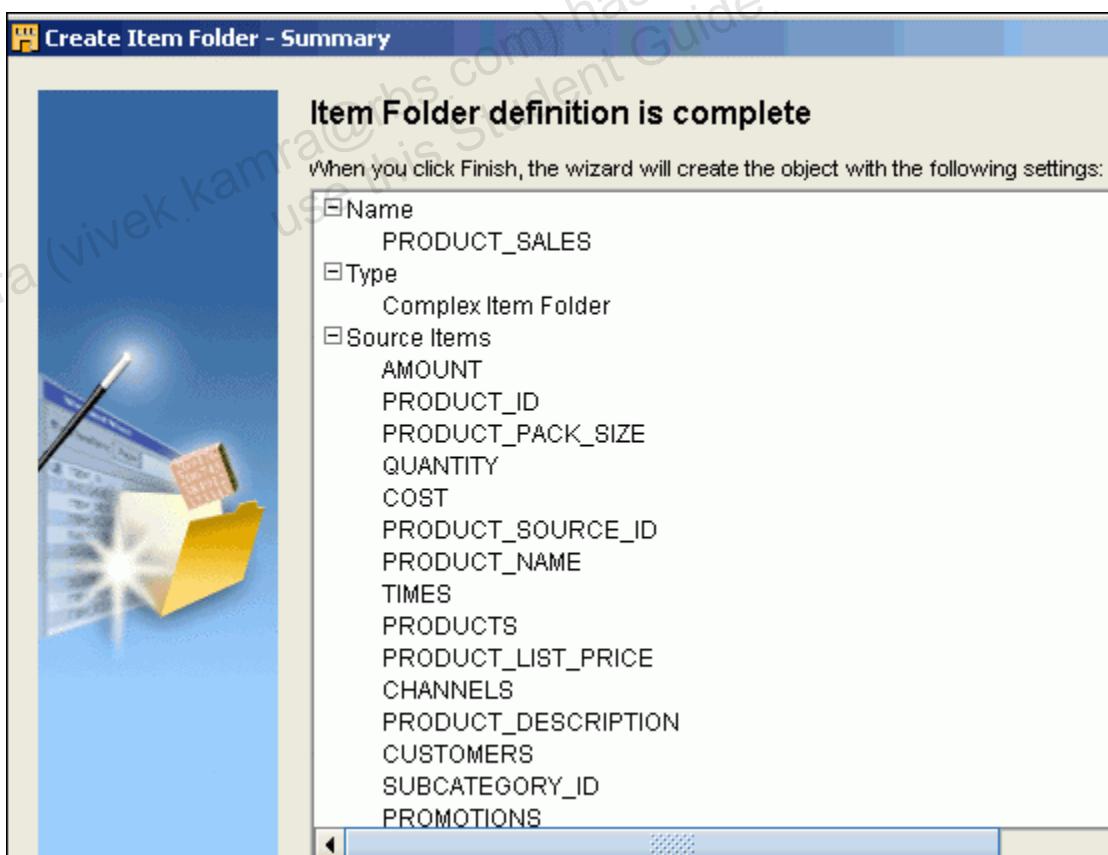


4. On the Source Items page, move the PRODUCTS item folder to the Selected column. Then move SALES to the Selected column. (You cannot multiselect items on this page.) Click Next.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



5. On the Summary page, note all the items selected from the two simple item folders for this complex item folder. Click Finish.

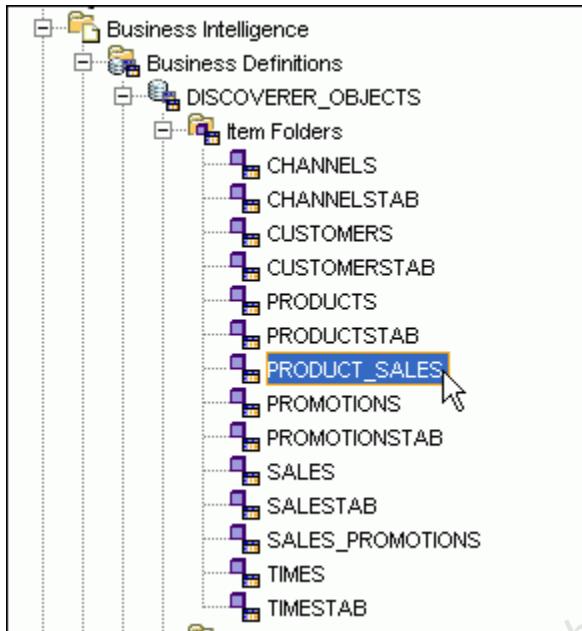


- 16) Open the PRODUCT_SALES item folder in the Data Object Editor.

Answer:

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

1. In the project Explorer, expand DISCOVERER_OBJECTS > Item Folders.
2. Double-click PRODUCT_SALES to open its editor.



- 17) For the AMOUNT item, choose AVG as the default aggregation.

Answer:

1. In the Item Folder Details panel at the bottom, click the Items tab, and then click the AMOUNT item.
2. In the Item Folder Details panel, scroll down till you see the Item Properties section. See the following screenshot. **Note:** Minimize the Canvas panel for better and expanded view of the Item Folder Details panel.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

The screenshot shows the 'Item Folder Details' panel for the 'PRODUCT_SALES' item folder. The 'Items' tab is selected. The 'Item Details' table lists seven items: AMOUNT, PRODUCT_ID, PRODUCT_PACK_SIZE, QUANTITY, COST, PRODUCT_SOURCE_ID, and PRODUCT_NAME. All items are visible to users and have descriptions. The 'Default aggregate' dropdown menu is open, showing options: NUMBER, SALES.AMOUNT, <null>, Measure, MAX, AVG, COUNT, Detail, MAX, MIN, SUM, and General. The 'MAX' option is selected.

	Name	Visible to user	Description
1	AMOUNT	<input checked="" type="checkbox"/>	Sales Amo...
2	PRODUCT_ID	<input checked="" type="checkbox"/>	
3	PRODUCT_PACK_SIZE	<input checked="" type="checkbox"/>	Pack Size
4	QUANTITY	<input checked="" type="checkbox"/>	Sales Quan...
5	COST	<input checked="" type="checkbox"/>	Sales Cost
6	PRODUCT_SOURCE_ID	<input checked="" type="checkbox"/>	Source ID
7	PRODUCT_NAME	<input checked="" type="checkbox"/>	Name

Specify the Item properties

General

Datatype: NUMBER
Formula: SALES.AMOUNT
Database Column: <null>
Item Class: Measure
Default position: MAX
Default aggregate: MAX
Heading: AVG
Format mask: COUNT
Alignment: Detail
Word wrap: MAX
Case storage: MIN
Display case: SUM
Default width: General
Replace NULL with:

3. With the AMOUNT item selected, change the default aggregate from SUM to MAX.
- 18) Add an item named PROFIT, with the “Visible to user” option selected.

Answer:

In the Item Folder Details panel, scroll up to the list of items. Add another item, PROFIT, with the “Visible to user” option selected.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

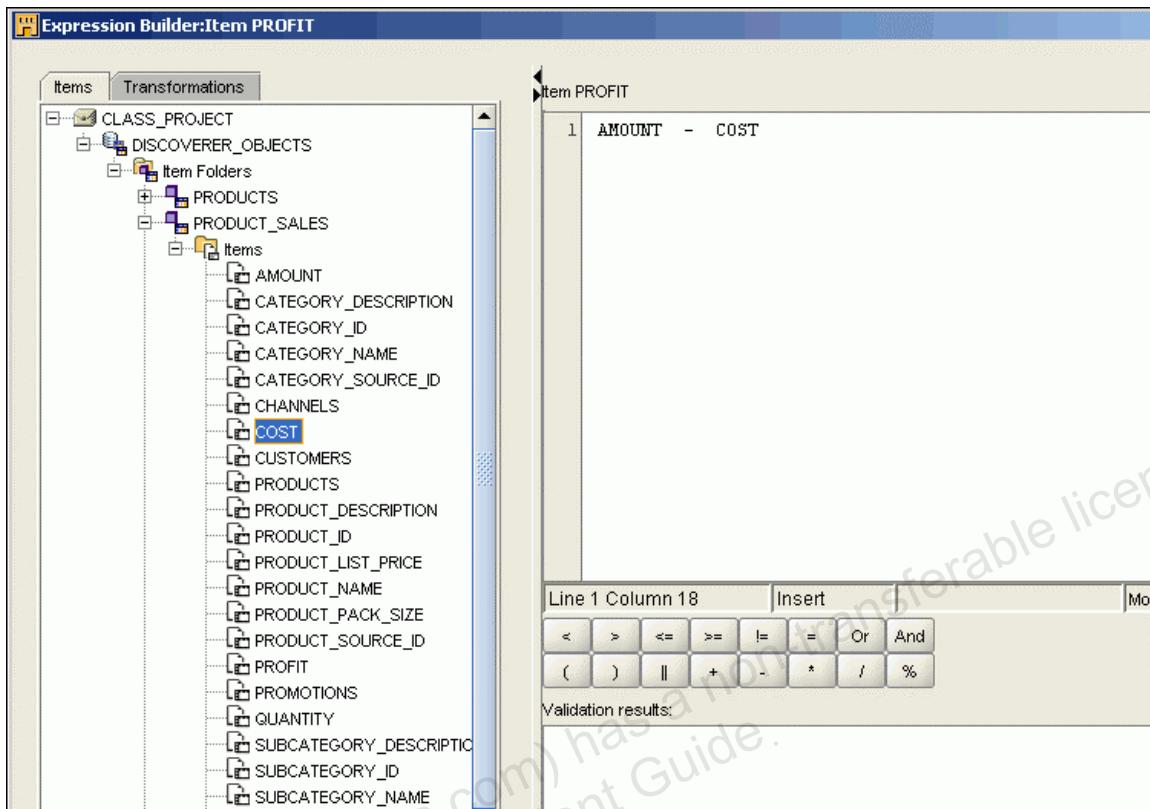
Item Folder Details: DISCOVERER_OBJECTS.PRODUCT_SALES "ReadWrite"			
	Name	Source Items	Items
		Joins	Conditions
Item Details:			
	Name	Visible to user	Description
22	CATEGORY_ID	<input checked="" type="checkbox"/>	
23	TOTAL_DESCRIPTION	<input checked="" type="checkbox"/>	Description
24	TOTAL_SOURCE_ID	<input checked="" type="checkbox"/>	Source ID
25	TOTAL_NAME	<input checked="" type="checkbox"/>	Name
26	TOTAL_ID	<input checked="" type="checkbox"/>	
27	PROFIT	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

- 19) Create a formula for this newly created PROFIT item. The formula is:
PRODUCT_SALES.AMOUNT – PRODUCT_SALES.COST

Answer:

1. If you have minimized the Canvas panel, maximize it now. On the canvas, select the PRODUCT_SALES item folder, and scroll down to the PROFIT item. Right-click the PROFIT item, and select Edit Formula. The Expression Builder appears.
2. In the Expression Builder, specify the formula: PRODUCT_SALES.AMOUNT – PRODUCT_SALES.COST. (The formula appears only as AMOUNT – COST unless you exit and reopen the Expression Builder.) Click OK.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



20) Save your work.

Answer:

In the Data Object Editor, click the Save All icon on the toolbar.

21) Close the Data Object Editor.

Answer:

In the Data Object Editor, from the Diagram menu, select Close Window.

Create a List of Values

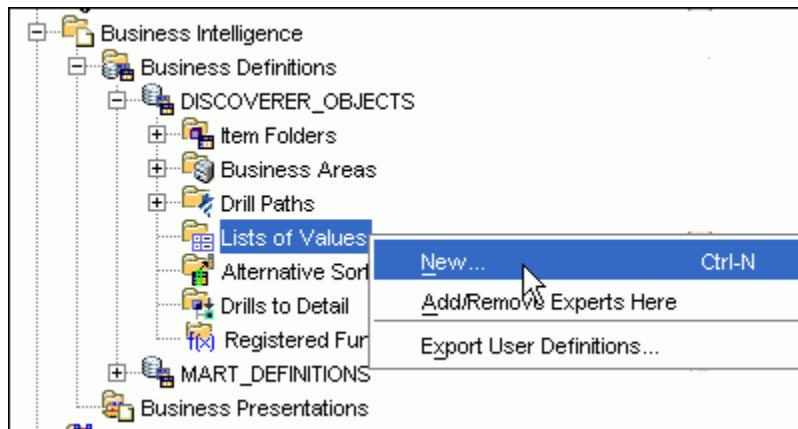
Your manager has an additional requirement: Show a list of values for all products, enabling the report to be restricted based on one or more items selected from the list of products. To implement this requirement, create a list of values.

22) Create a list of values named PRODUCTS_LOV. Select PRODUCTS.PRODUCT_NAME as the item that defines the list of values. Select PRODUCTS_SALES.PRODUCT_NAME as the item that references the list of values.

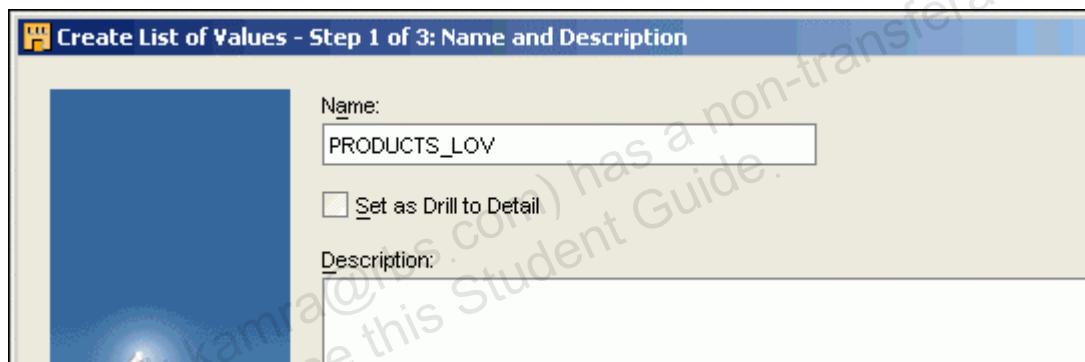
Answer:

1. In the Project Explorer, expand **DISCOVERER_OBJECTS**. Right-click the **Lists of Values**, and select **New**.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

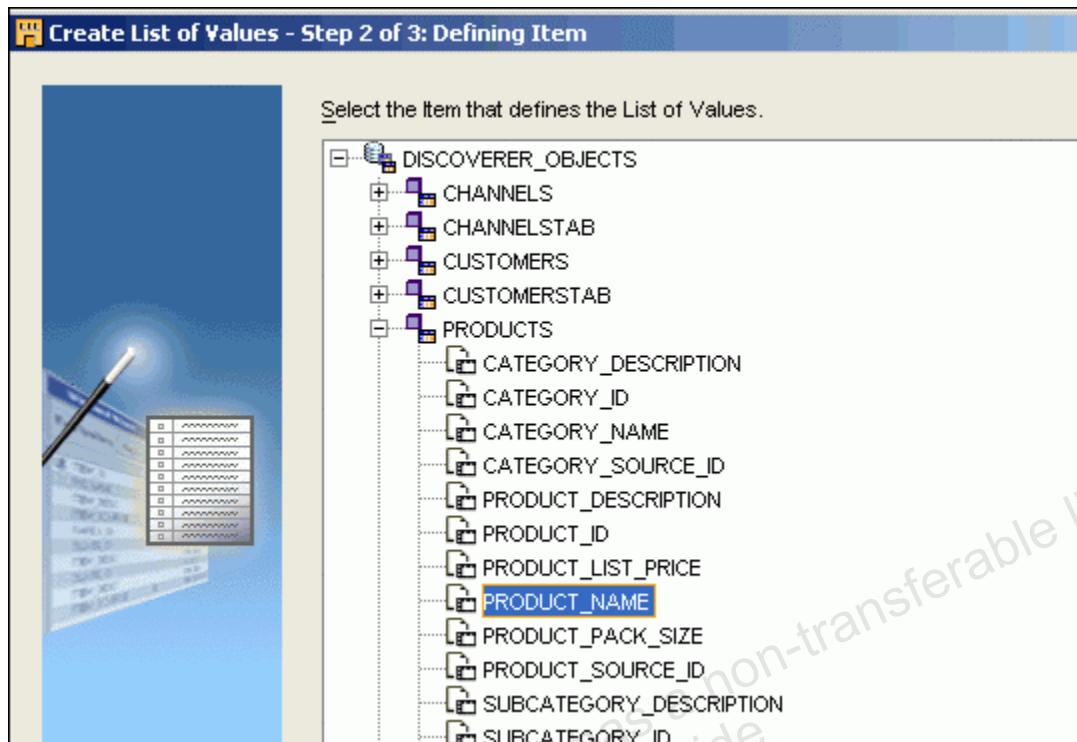


2. On the Name and Description page, name the list of values as PRODUCTS_LOV. Leave “Set as Drill to Detail” deselected. You come back to this after creating the list of values. Click Next.



3. On the Defining Item page, expand the PRODUCTS folder, and select PRODUCT_NAME as the item that defines the list of values. This is shown in the next screenshot.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)



4. Click Next. On the Referencing Items page, note that the defining item PRODUCT_NAME in the PRODUCTS folder is already added to the list of items that use the list of values. Select the PRODUCT_NAME item from the PRODUCT_SALES folder, and shuttle it to the Selected list.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Create List of Values - Step 3 of 3: Referencing Items

Select the Items that reference the List of Values.

Available:

- +
- CUSTOMERS
- CUSTOMERSTAB
- +
- PRODUCTS
- PRODUCTSTAB
- +
- PRODUCT_SALES
 - AMOUNT
 - CATEGORY_DESCRIP
 - CATEGORY_ID
 - CATEGORY_NAME
 - CATEGORY_SOURCE
 - CHANNELS
 - COST
 - CUSTOMERS
 - PRODUCTS
 - PRODUCT_DESCRIPTION
 - PRODUCT_ID
 - PRODUCT_LIST_PRICE
 - PRODUCT_NAME
 - PRODUCT_PACK_SIZE

Selected:

- DISCOVERER_OBJECTS
 - PRODUCTS
 - PRODUCT_NAME

Add (ALT-D)

Select the Items that reference the List of Values.

Available:

- PRODUCT_SALES
 - AMOUNT
 - CATEGORY_DESCRIP
 - CATEGORY_ID
 - CATEGORY_NAME
 - CATEGORY_SOURCE
 - CHANNELS
 - COST
 - CUSTOMERS
 - PRODUCTS
 - PRODUCT_DESCRIPTION
 - PRODUCT_ID
 - PRODUCT_LIST_PRICE
 - PRODUCT_PACK_SIZE
 - PRODUCT_SOURCE_ID
 - PROFIT
 - PROMOTIONS
 - QUANTITY
 - SUBCATEGORY_DESCRIP

Selected:

- DISCOVERER_OBJECTS
 - PRODUCTS
 - PRODUCT_NAME
 - PRODUCT_SALES
 - PRODUCT_NAME

5. Click Next. Review the summary, and click Finish.

Solutions for Practice 11-1: Defining Business Intelligence Objects Based on Data Objects (continued)

Create “Drill to Detail”

Now your manager wants to drill down to detail on the specifics of the product. To enable this functionality for an item, you can create Drill to Detail and relate the item to that Drill to Detail.

Currently, each of the following constructs materializes as an item class in Discoverer:

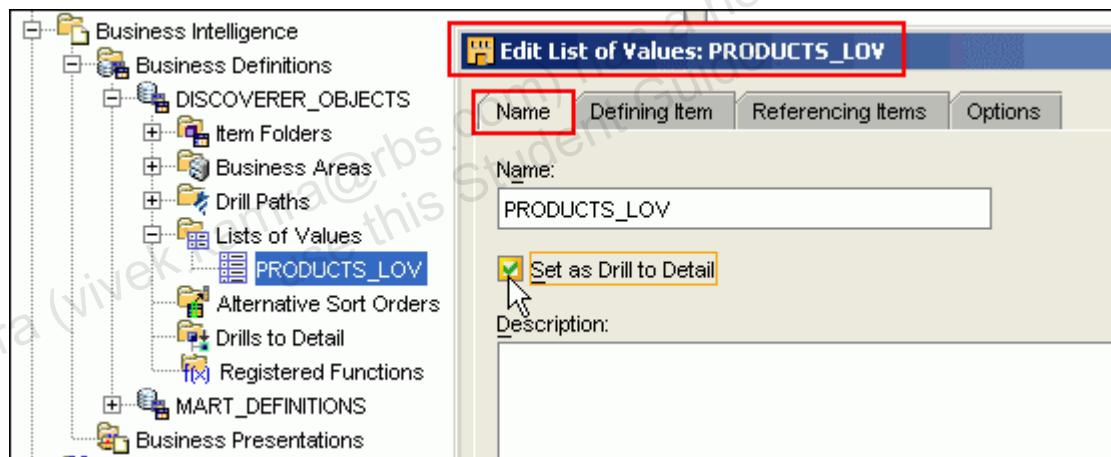
- List of Values
- Drill to Detail
- Alternative Sort

Discoverer allows only a maximum of one item class per item. Therefore, to enable Drill to Detail on the item class that implements your PRODUCTS_LOV list of values, you need to edit the PRODUCTS_LOV list of values.

23) Edit PRODUCTS_LOV to select the Set As Drill to Detail option.

Answer:

1. Double-click the PRODUCTS_LOV list of values.



2. On the Name tabbed page, select Set as Drill to Detail. Click OK.

24) Save your work.

Answer:

Click the **Save All** icon on the toolbar.

Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer

In the previous practice, you defined business intelligence objects for Discoverer. In this practice, you use Discoverer Administrator to create an End User Layer (EUL), deploy the OWB business intelligence objects to the End User Layer schema, and then use Discoverer Plus Relational to analyze the data.

In Discoverer Administrator, Create an EUL in the DISCO Database User

- 1) Using Oracle Discoverer Administrator, create an EUL named DISCO.

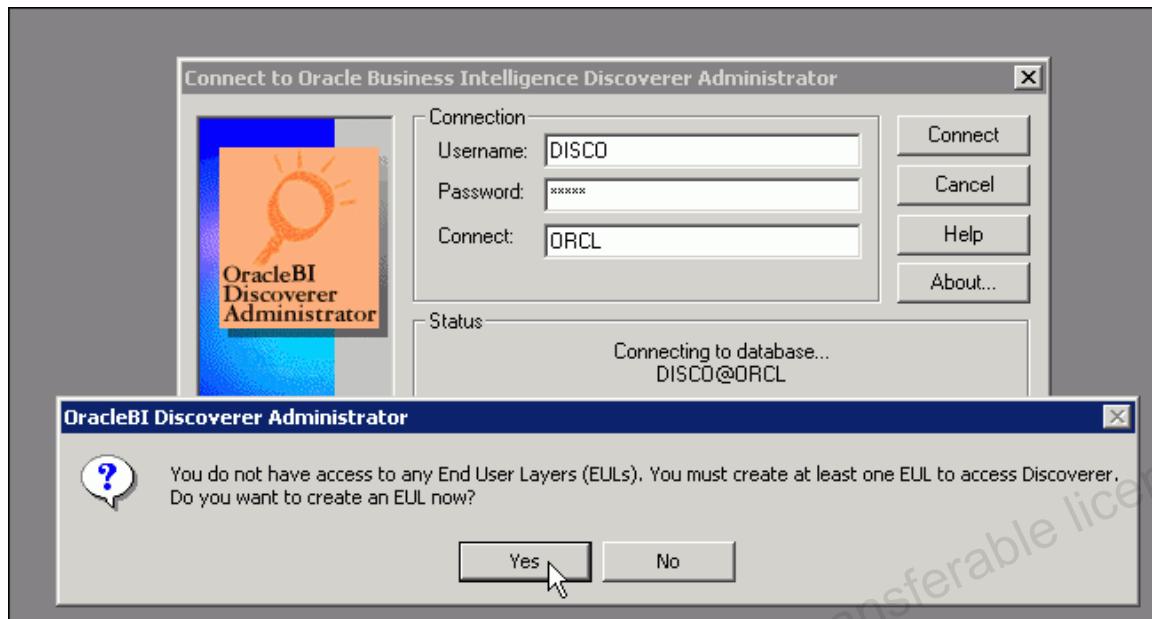
Answer:

1. Double-click the Discoverer Administrator Desktop icon (or Start > Programs > Oracle Business Intelligence Tools- BIToolsHome > Oracle Discoverer Administrator).
2. In the Connection dialog box, enter username and password as DISCO. Click Connect.

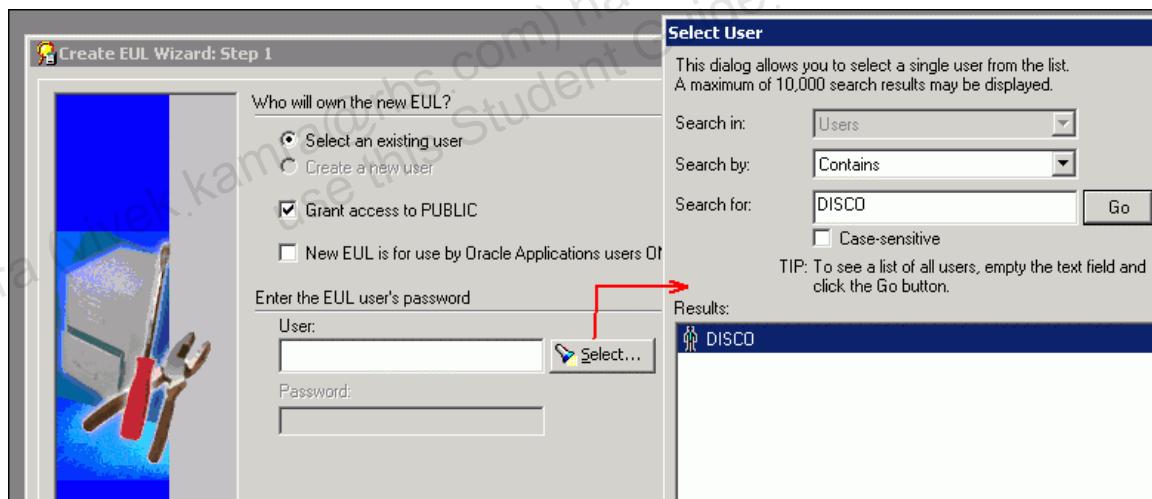


3. You receive a message asking whether you want to create an EUL now. Click Yes.

Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)



4. In the EUL Manager dialog box, click Create a EUL.
5. On the Create EUL wizard's Step 1 page, accept the defaults, and click Select.



6. In the Select User dialog box, enter DISCO in the "Search for" field, and click Go. DISCO appears in the Results field. Click OK. Click Finish. A progress bar appears, and eventually a message appears indicating that the EUL has been created. Click OK. Click Close.
7. A Load Wizard appears prompting you to create or open a business area. However, you instead deploy a business area to this EUL using OWB. So click Cancel and exit Discoverer Administrator.

Deploy BI Objects to the End User Layer

You deploy the BI objects to the End User Layer.

- 2) Register DISCO_LOCATION.

Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)

Answer:

1. In the Control Center Manager, on the left, right-click DISCO_LOCATION, and select Register. In the pop-up window, click Test Connection. When the results indicate success, click OK.
- 3) Deploy DISCOVERER_OBJECTS to the EUL.

Answer:

1. In the Control Center Manager, select DISCOVERER_OBJECTS.
2. In the Object Details panel, click **Default Actions**.
3. On the toolbar, click **Deploy**. A new Job entry appears in the Control Center jobs panel at the bottom of the page. Its Status symbol initially represents Generation. This should then change to a green arrow (Running), and finally a check mark.

Grant Select Privileges to DISCO

- 4) Grant select privileges to DISCO on tables in SALES_WH.

Answer:

1. Log in to SQL*Plus as sales_wh/sales_wh.
2. At the SQL prompt, enter @e:\labs\sql\grants_on_sales.sql.

Log In to Discoverer Plus Relational as EUL_FROM_OWB and Create a Worksheet

- 5) Open OracleBI Discoverer to create a worksheet from the populated EUL.

Answer:

1. Double-click the Start BI Middle Tier desktop icon to start the services (ProcessManager) needed by Discoverer Plus Relational. These services may take several minutes to open.
2. When the window showing the services disappears, double-click the Disco_Plus desktop icon. It may take several minutes for the browser connection page to appear.
3. When the Connection page appears, enter the following information:
Connect To: OracleBI Discoverer
User Name: disco
Password: disco
Database: orcl
End User Layer: disco
Click Go.

Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)

The screenshot shows the Oracle BI Discoverer Plus interface. At the top, it says "Discoverer Plus" and "Business Intelligence". Below that, a section titled "Connect to OracleBI Discoverer" displays the message "Welcome to OracleBI Discoverer.". Underneath, there's a sub-section titled "Connect Directly" with the instruction "Enter your connection details below to connect directly to OracleBI Discoverer.". It includes a note "* Indicates required field." and a form with the following fields:

Connect To	OracleBI Discoverer
* User Name	disco
* Password	*****
* Database	orcl
End User Layer	disco
Locale	Locale retrieved from browser

At the bottom of the form is a "Go" button.

Log In to Discoverer Plus Relational as DISCO and Create a Worksheet

If you receive a JInitiator Security Warning pop-up message, click “Grant this session.”

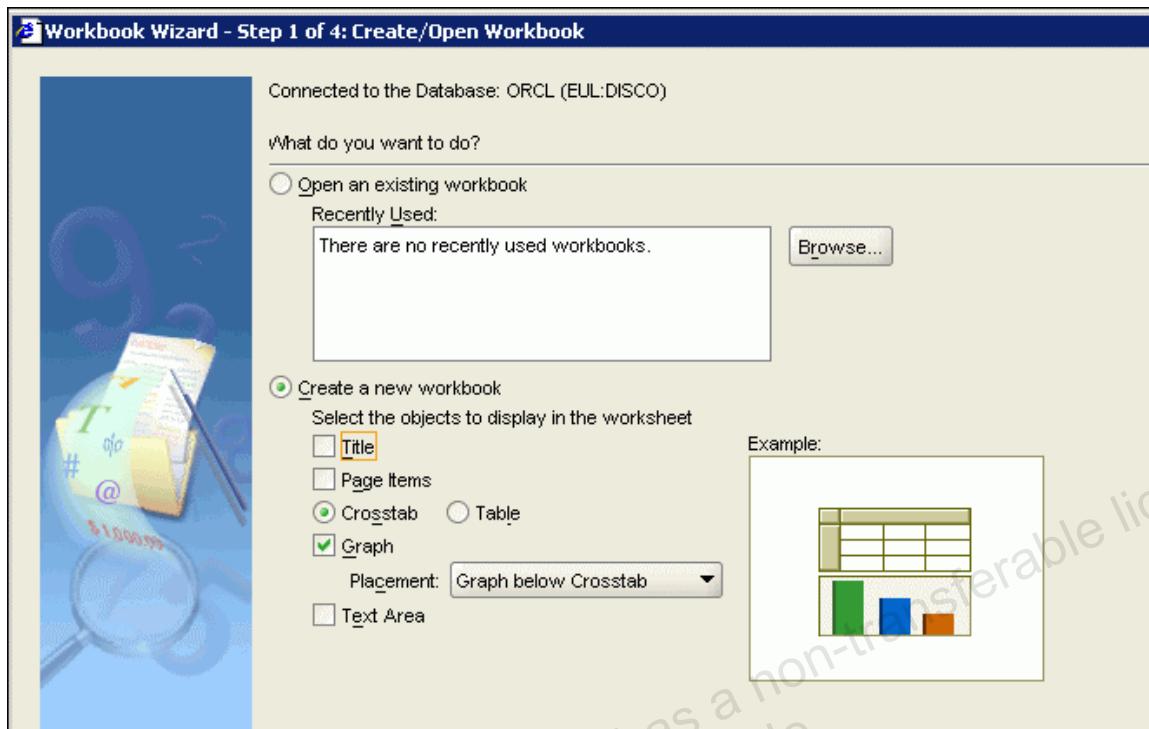
The Workbook Wizard is launched.

- 6) On the Workbook Wizard, select Crosstab and Graph and deselect all other check boxes.

Answer:

On the Create/Open Workbook page, “Create a new workbook” is selected. Select Crosstab and Graph, and deselect all other check boxes. Click Next.

Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)

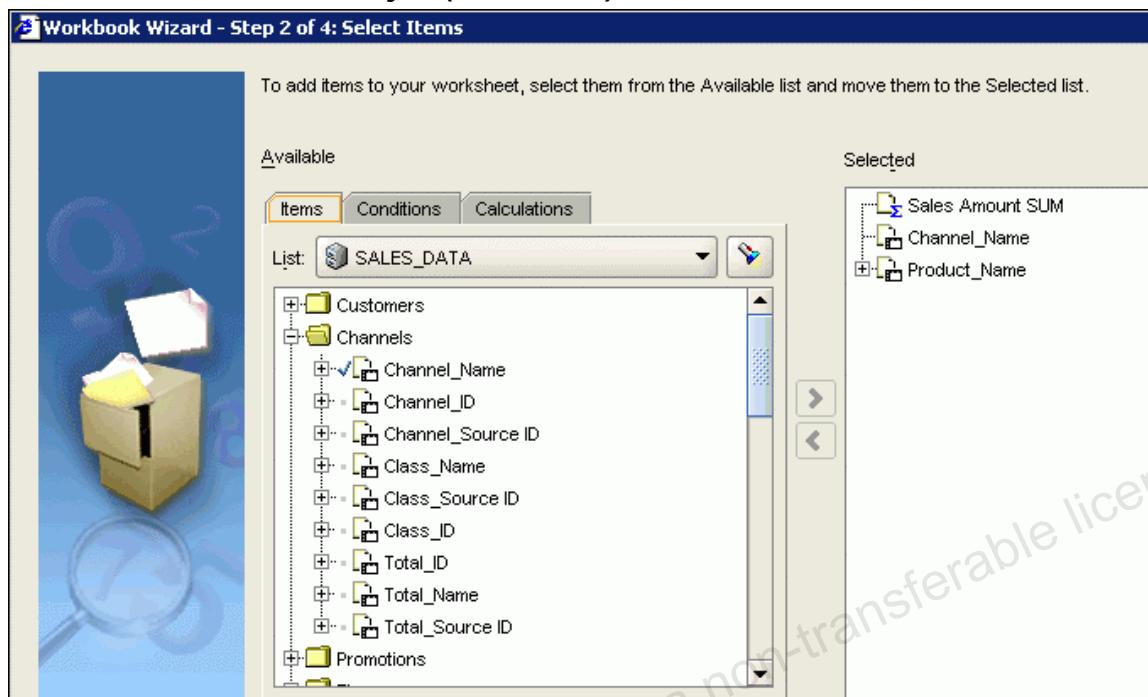


- 7) On the Select Items page, select the following: PRODUCTS.Product_Name, CHANNELS.Channel_Name, and SALES CUBE.Sales_Amount.

Answer:

On the Select Items page, expand PRODUCTS, select Product_Name, and move it to the Selected list. Expand SALES CUBE, select Sales_Amount, and move it to the Selected list. Expand Channels, select Channel_Name, and move it to the Selected list. Click Next.

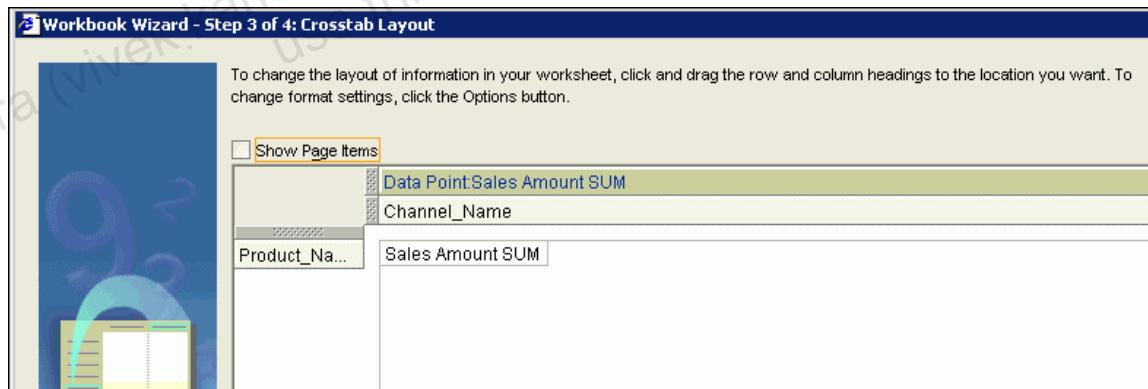
Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)



- 8) On the Crosstab Layout page, arrange the items as shown in the screenshot.

Answer:

On the Crosstab Layout page, you can arrange the items by moving them around. Arrange the items as shown in the screenshot:



Make sure that you have at least one side item to run down the left. Click Finish. Now you can analyze the data.

Solutions for Practice 11-2: Deploying OWB Business Intelligence Objects to a Discoverer End User Layer (continued)

Product_Name	Channel_Name	Sales Amount SUM		
		► Direct Sales	► Internet	► Partners
► 1.44MB External 3.5" Diskette		138170.41	22255.76	60503.14
► 128MB Memory Card		315483.78	89251.24	169500.41
► 17" LCD w/built-in HDTV Tuner		4457699.64	1058952.3	1692366.8
► 18" Flat Panel Graphics Monitor		3038192.49	1092383.31	1127568.55
► 256MB Memory Card		394743.84	71203.21	127465.48
► 3 1/2" Bulk diskettes, Box of 100		251148.22	31545.15	89103.61
► 3 1/2" Bulk diskettes, Box of 50		158808.74	26181.25	70499.44
► 5MP Telephoto Digital Camera		3400862.36	1381685.55	1561476.75
► 64MB Memory Card		13839.95	4767.1	9489.93
► 8.3 Minitower Speaker		1736923.68	1007545.07	1107740.14
► Adventures with Numbers		114737.73	18077.88	43456.27
► Bounce		125650.53	38860.01	81562.42
► CD-R Mini Discs		259168.4	37037.77	89793.5
► CD-R with Jewel Cases, pACK OF 12		95958.63	20436.87	55277.7
► CD-R, Professional Grade, Pack of 10		108998.31	17901.45	44458.22
► CD-RW, High Speed Pack of 5		144882.36	23670	59975.9
► CD-RW, High Speed, Pack of 10		69847.16	12010.24	25233.48
► Comic Book Heroes		76484.91	8671.24	16494.27
► DVD-R Disc with Jewel Case, 4.7 GB		311400.09	131909.96	136008.46
► DVD-R Discs, 4.7GB, Pack of 5		580120.97	99509.09	229076.03

MOLAP: Modeling and Loading

Appendix C

ORACLE

Copyright © 2006, Oracle. All rights reserved.

Objectives

After completing this lesson, you should be able to describe how Warehouse Builder can do the following:

- **Model multidimensional dimensions and cubes, and their mappings**
- **Directly deploy an OLAP cube to the analytic workspace**
- **Directly load an OLAP cube to the analytic workspace**
- **Use the Repository Browser to monitor your loads**
- **View loaded cube data, including pivot and drill down**



Copyright © 2006, Oracle. All rights reserved.

What Is OLAP?

OLAP stands for *online analytical processing*.

- **Online:** You have access to live data (rather than static data).
- **Analytical:** You can analyze your data for simple reporting. You can create reports that are:
 - Multidimensional
 - Calculation intensive
 - Supported by time-based analysis
 - Ideal for applications with unpredictable, ad hoc query requirements



Copyright © 2006, Oracle. All rights reserved.

What Is OLAP?

Online analytical processing (OLAP) is a term that has been used since the early 1990s to describe a class of computer systems designed and optimized for analysis. By using this term, you can differentiate the requirements of the data analyst from the requirements of the users of online transaction processing (OLTP). In the context of business intelligence today, the emphasis is more on “online” and “analytical.”

Online: Although most OLAP tools and applications enable development of reports that can be saved and printed when not connected to live data, OLAP emphasizes live access to data rather than static reporting. Analytic queries are submitted against the database in real time, and the results are returned to your computer screen.

Analytical: This is the key concept with OLAP. End users can:

- Easily navigate multidimensional data to perform unpredictable ad hoc queries and to display the results in a variety of interesting layouts
- Drill down through levels of detail to uncover significant aspects of the data
- Rapidly and efficiently obtain the results of sophisticated data calculation and selection across multiple dimensions of the data

Who Needs OLAP Data?



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Who Needs OLAP Data?

OLAP data, which can provide answers to an organization's specific business questions, is valuable to both IT (information technology) users and end users alike. However, different user communities have different reporting and analysis requirements. They approach the various business data from many perspectives.

Here are some examples:

- A typical financial perspective on the data is to look at revenues, costs, cash flows, and similar measures (factual data) for a particular time frame—perhaps the end of the month, the quarter, or the year. Comparisons with budgets and forecasts are also common.
- Marketing analysts commonly look at sales performance for groups of products and various customer segments across time. They may do this in the context of comparing information with competitors and other market information that is available to their organization. This external information may come from sources such as third-party market research firms.
- Sales representatives are typically interested in analyzing their customers on a regional basis—for all products—with respect to a specified sales target.

Typical Business Questions

- **How do sales for our five most profitable products across the U.S., for this quarter, compare with sales a year ago?**
- **What are the differences in the product-sales mix between the regions, relative to the global sales mix?**
- **What are our forecast units, unit price per service, unit cost per product, sales, cost trends, and profit for the next 12 months?**
- **In what ways does the mix vary by salesperson, and what is the relative performance of our salespeople?**
- **What are the products making up 40% of our profit for each region over time?**



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Typical Business Questions

A standard transactional report or query might ask, “When was order 84305 shipped?” This query reflects the basic mechanics of doing business. It involves simple data selection and little or no calculation processing.

In contrast, OLAP systems are typically deployed to extend and enhance an organization’s ability to answer a much broader range of business questions about data (examples of which are presented in the slide). These questions are more analytical and complex, and the answer to one question often leads immediately to another question as the user follows a train of thought in researching a business problem or opportunity.

OLAP is designed to make it easy for end users to ask these types of analytical questions without requiring:

- Assistance from IT
- Programming skills
- Technical knowledge about the organization of the database

The results of queries also need to be rapid so that the analyst’s train of thought is not interrupted and the value of the analysis is not diminished.

Examining an OLAP Question

An OLAP question is a multidimensional query, as in the following:

“What was the percentage change in revenue for a grouping of our top 20% products, over a rolling three-month time period this year compared to the same period last year, for each region of the world?”



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Examining an OLAP Question

The questions that business users tend to ask are naturally multidimensional. They use a multidimensional language to express the business questions that they ask (such as the one shown in the slide).

OLAP Questions Are Multidimensional Queries

The OLAP question shown in the slide is a common example of a multidimensional query. It describes both the data that the user wants to examine and the structural form of that data. Business users typically want to answer questions that include terms such as *what*, *where*, *who*, and *when*. For example, you find the following essential questions embedded in the sample question:

- *What products are selling best?* (“...top 20%...”)
- *Where are they selling?* (“...each region of the world...”)
- *When have they performed the best?* (“...percentage change in revenue...”)

Examining an OLAP Question

“What was the percentage change in revenue for a grouping of our top 20% products, over a rolling three-month time period this year compared to the same period last year, for each region of the world?”

This is a simple business question, but the actual query can be quite complex.

- There are two calculations (% change in revenue and rolling three-month total).
- There is a ranking element (the top 20% of products).
- There are aggregations (region level of the geographic dimension).
- There are multidimensional selections (specific products, specific time periods, and specific regions).
- The result of the query is a multidimensional view of the data (perhaps a tabular display on screen, perhaps as a graph, or both).
- The complexity of this query is a technical issue rather than a business issue.
- Therefore, a key goal of OLAP technology is to make it very easy for end users to ask such questions.



Copyright © 2006, Oracle. All rights reserved.

Examining an OLAP Question (continued)

A Simple Business Question: A Complex Query

There is much complexity contained in this query.

Many would categorize it as quite a complex query. And, for many traditional query and reporting tools and databases, it is quite complex indeed. But the complexity of this query is a *technical* issue rather than a *business* issue.

Therefore, a key goal of most OLAP technology is to make it very easy for end users to ask such questions about their data without placing a burden on the IT department.

Managing Multidimensional Queries Without OLAP

Your users need to build reports with:

- Units for a prior period
- Change in units from the prior period
- Share of units by product
- Moving average of units
- Cost
- Percentage change in cost from the prior period
- Share of cost by product
- Moving average of cost
- Sales
- Sales for a prior period
- Change in sales from the prior period
- Percentage change in sales from the prior period
- Profit
- Profit for a prior period
- Percentage change in profit from the prior period
- Share of profit by product



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Managing Multidimensional Queries Without OLAP

People invest in OLAP technology primarily because of its ability to quickly and efficiently generate calculated measures for use in end-user queries and reports.

Suppose that you are required to provide an online reporting and analysis system for the end users and colleagues across your organization without OLAP. Your data warehouse or operational system provides three stored measures: Sales Units, Price, and Cost. You know that your end users need to build their reports and queries with all these calculated measures.

Managing Multidimensional Queries Without OLAP

- **Would you:**
 - Try to build these measures in your ETL tool
 - Create SQL functions
 - Define calculations in the reporting tool
- **How long would your users have to wait for the system to be built?**
- **How would the system perform?**

ORACLE®

Copyright © 2006, Oracle. All rights reserved.

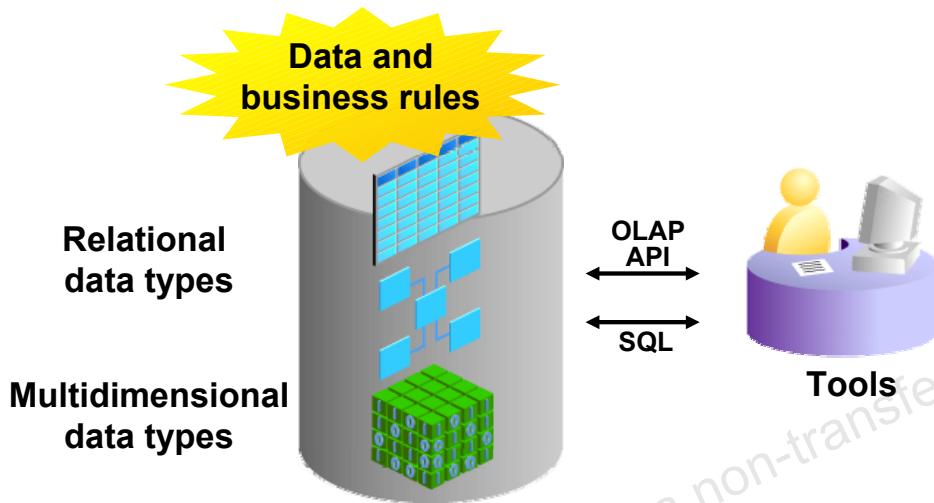
Managing Multidimensional Queries Without OLAP (continued)

Without OLAP, most organizations would have some difficult choices to make. They might:

- Have their IT department build additional programming into the extract, transform, and load (ETL) processing that gets the data out of the source systems and loads it into the reporting database. In this way, some of the additional numbers can be precalculated and stored in the database.
- Create special SQL functions that allow the calculations to be accessed from a tool but processed at run time in the database
- Ask end users to define the calculations themselves in the query and reporting tool
- Decide to reduce the functionality of the system to make it easy and inexpensive to deploy, but risk not delivering the full business benefit to end users

If the IT department decides to build the system, how long would it take to deliver the reports that the users asked for? How fast would the system perform, in terms of both query performance and batch processing performance, when preparing the data for the user?

Managing Multidimensional Queries with Oracle OLAP



ORACLE®

Copyright © 2006, Oracle. All rights reserved.

Managing Multidimensional Queries with Oracle OLAP

Oracle OLAP has an advanced multidimensional OLAP capability inside the main database, the place where your data is collected and securely stored and managed. With such a capability, you benefit from the following:

- All the data is in one place.
- All the business rules are defined and stored in one place.
- All the data is secured and managed in the same place, the Oracle database.

Native Access Options

The OLAP option in the Oracle database provides a sophisticated OLAP API that is used by those Oracle business intelligence tools that are aware of the dimensional model. These dimensionally aware OLAP tools can access both multidimensional and relational data types.

In addition to its native OLAP API, the Oracle database also provides SQL access to both relational and multidimensional data types that are provided by the OLAP option.

Dimensional Modeling

- **Customers should need to think only about designing the dimension and cube, and not about how the Oracle database stores data and performs ETL.**
- **Customers want to focus on product hierarchy; the Oracle database will figure out how and where to store data and perform ETL.**



ORACLE

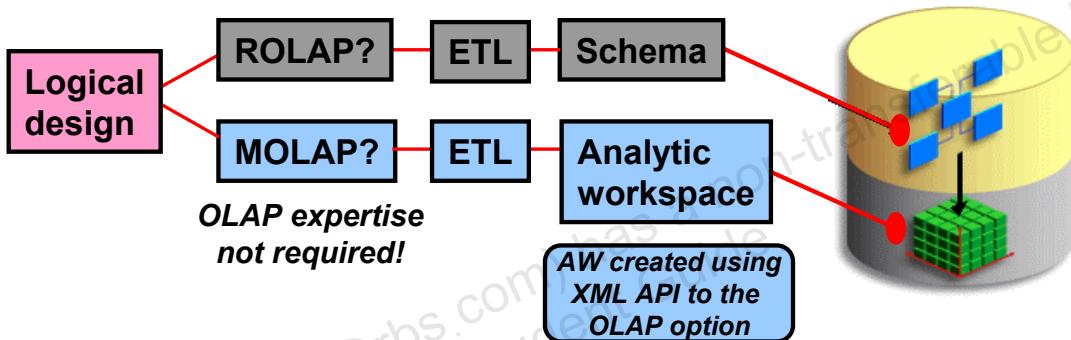
Copyright © 2006, Oracle. All rights reserved.

Dimensional Modeling

Because there are two separate tasks identified (defining the dimension and implementing the warehouse in a database), Oracle Warehouse Builder (also called as Warehouse Builder or OWB) splits these tasks. However, by allowing the implementation strategy to be achieved with a single click, Warehouse Builder makes it easy to use.

Enabling OLAP Solutions

- *OWB 10g Release 2 is capable of directly loading any data into the analytic workspace, enabling for the first time the transformation power on OLAP data loads.*
- Create a logical design describing your OLAP cubes in dimensions, hierarchies, measures, calculated measures, and other components.
- Warehouse Builder uses the new XML API to the OLAP option to directly create the analytic workspace, its objects, and the metadata required in the database catalogs.
- After you have created your cubes and dimensions, use the Warehouse Builder ETL modelers to create the load programs, independent of storage decisions.



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Enabling OLAP Solutions

With Warehouse Builder, you can also choose your implementation. Do you want a pure multidimensional OLAP solution or a relational OLAP solution? With the click of a button, you can decide or change your mind if testing points you to a different direction.

Allowing you to load data from any source directly into the arena of high-end analytics of the OLAP option brings high-end analytics to the masses allowing many people to access and work with OLAP data.

OLAP Expertise Not Required

You need concern yourself only with such things as granularity, which summaries you want, which levels you want, and so on. You need not know the OLAP Structured Programming Language (SPL).

You need focus only on your logical design; Warehouse Builder transparently handles physical implementation.

Enhanced OLAP Support in OWB 10g R2

Benefits:

- **Productivity: Direct OLAP deployment**
- **Performance: Direct multidimensional loading**
- **Integration:**
 - **Seamless integration within process flows and dependency management**
 - **Fully functional, dimensionally aware data viewers, enabling you to inspect data from within Warehouse Builder**
 - **Lineage and impact analysis for Oracle OLAP objects**
- **Completeness: Full OLAP support**

ORACLE

Copyright © 2006, Oracle. All rights reserved.

Enhanced OLAP Support in OWB 10g R2

On an ongoing basis, Oracle Warehouse Builder tries to support as many database features as possible. The 9.0.4 release of Warehouse Builder introduced the first support for OLAP. It was slightly enhanced in the 9.2 version.

The biggest benefit of an OLAP solution is the capability to deliver highly advanced calculation capabilities with amazing response times to end users. Now that the Oracle OLAP option is available inside the Oracle database, Oracle Warehouse Builder 10g Release 2 enables this technology for you in an easy-to-use way.

There will be direct object deployment into OLAP in the same way as there is relational object deployment today. Also, rather than loading data via the relational environment, there is support for direct data loading into OLAP.

Some of the OLAP features (such as aggregations and custom measures) that currently can be implemented using extensibility features are supported readily in Oracle Warehouse Builder 10g Release 2.

Differences Between OLAP and Relational Loading

- There are not many differences (from the user interface perspective) between OLAP cube loading and relational table loading.
- These steps remain the same in OLAP or relational deployment:
 1. Deploy the data objects (dimensions and cubes).
 2. Deploy the mappings.
 3. Run the mappings to load data into an OLAP cube.

ORACLE

Copyright © 2006, Oracle. All rights reserved.

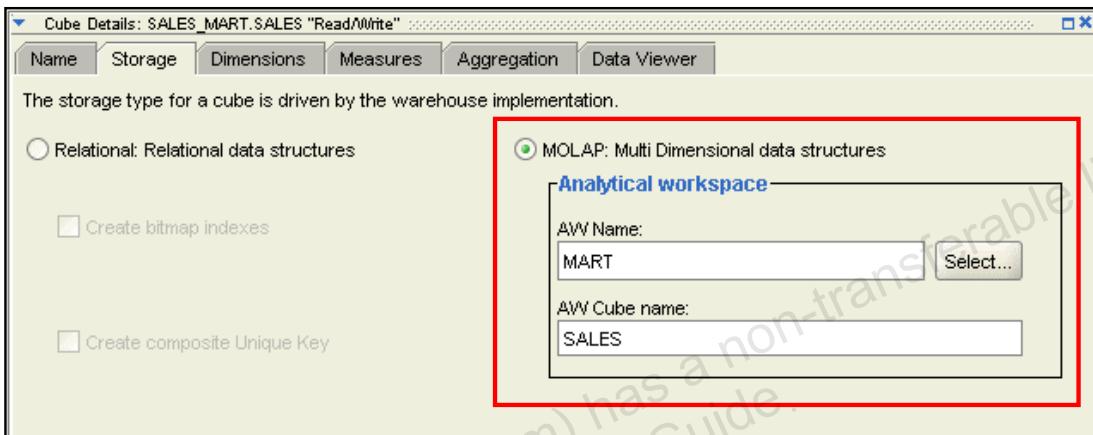
Differences Between OLAP and Relational Loading

From the user interface perspective, there are no differences between OLAP cube loading and relational table loading. You draw a mapping and its lines; Warehouse Builder creates the code.

In Oracle Warehouse Builder 10g Release 2, you create a logical design describing your OLAP cubes in dimensions, hierarchies, measures, calculated measures, and all the components that you need. Then Warehouse Builder uses the new XML API with the OLAP option to create the analytic workspace and the metadata required in the database catalogs.

Checking the Multidimensional Storage Type

Before deploying, inspect the information on the Storage tabbed page for the cube.



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Checking the Multidimensional Storage Type

For the storage type, check whether “MOLAP: Multi Dimensional data structures” is selected.

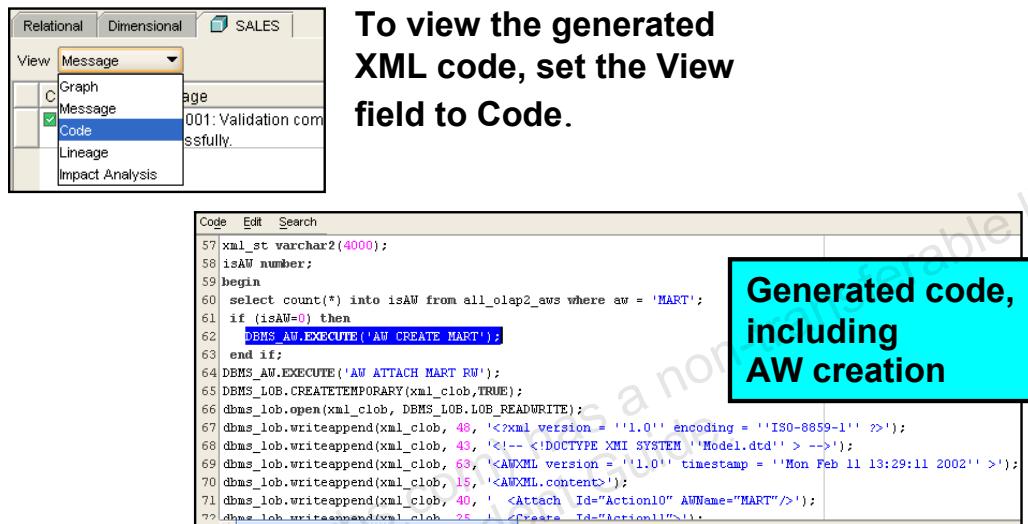
Relational or MOLAP Implementation

With Warehouse Builder, you can also choose your implementation. Do you want a pure multidimensional OLAP (MOLAP) solution or a relational OLAP solution? With a click of a button, you can decide or change your mind if testing points you to a different direction.

Certainly, you base this implementation decision upon careful needs assessment and design decisions.

Generating the XML Code

Before deploying, you can preview the XML code required for the Oracle 10g OLAP API by using the Generate facility for a dimension or cube.



The screenshot shows the Oracle Warehouse Builder interface. A context menu is open over a dimension named 'SALES'. The 'View' option is highlighted, with 'Code' selected. The main pane displays generated PL/SQL code for creating an analytic workspace (AW) and attaching it to the dimension. A callout box highlights the generated code, specifically the AW creation portion.

```
Code Edit Search
57 xml_st varchar2(4000);
58 isAW number;
59 begin
60 select count(*) into isAW from all_olap2_aw where aw = 'MART';
61 if (isAW=0) then
62   DBMS_AW.EXECUTE('AW CREATE MART');
63 end if;
64 DBMS_AW.EXECUTE('AW ATTACH MART RM');
65 DBMS_LOB.CREATETEMPORARY(xml_clob,TRUE);
66 dbms_lob.open(xml_clob, DBMS_LOB_READONLY);
67 dbms_lob.writeappend(xml_clob, 46, '<?xml version = "1.0" encoding = "ISO-8859-1" ?>');
68 dbms_lob.writeappend(xml_clob, 43, '<!--<!DOCTYPE XMI SYSTEM ''Model.dtd'' -->');
69 dbms_lob.writeappend(xml_clob, 63, '<?XML version = "1.0" timestamp = "Mon Feb 11 13:29:11 2002" ?>');
70 dbms_lob.writeappend(xml_clob, 19, '<?XML.content>');
71 dbms_lob.writeappend(xml_clob, 40, ' <Attach Id="Action10" AWName="MART"/>');
72 dbms_lob.writemandxml(xml_clob, 25, '<Create Td="Action11"/>')
```

Generated code, including AW creation

ORACLE

Copyright © 2006, Oracle. All rights reserved.

Generating the XML Code

Before deploying, you can preview the XML code required for the 10g OLAP API by using Warehouse Builder's Generate facility. The screenshot in the lower portion of the slide shows the generated code, which includes analytic workspace creation.

You see XML tags wrapped in database calls.

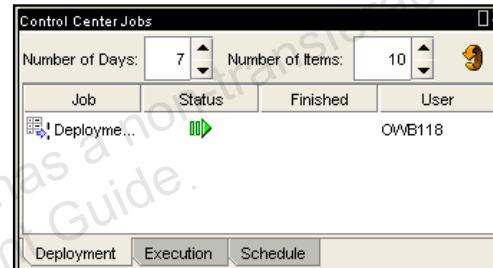
Deploying Data Objects with the Control Center Manager

LOAD_SALES	New	None	Not Deployed
CHANNELS	New	Create	Not Deployed
PRODUCTS	New	Create	Not Deployed
SALES	New	Create	Not Deployed
CLASS_DIM_SEQ	New	Create	Not Deployed
PROD_DIM_SEQ	New	Create	Not Deployed
SALES_MART_TO_XW...	New	Create	Not Deployed

Deploy



Control Center job monitor



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Deploying Data Objects with the Control Center Manager

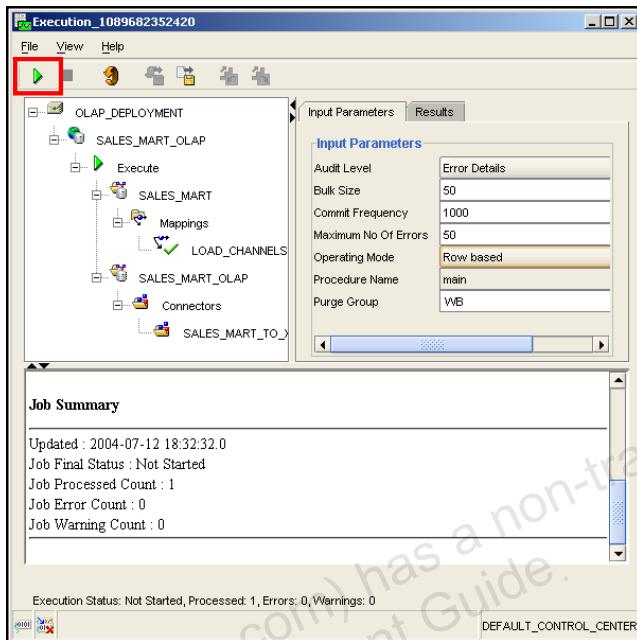
As with relational objects, you set the deployment action to “Create,” and deploy data objects by using the Control Center Manager.

Direct Loading into the Analytic Workspace

After you have created your schema or analytic workspace, you can use the Warehouse Builder ETL modelers to create the load programs. Oracle Warehouse Builder 10g Release 2 is now capable of *directly* loading any data into the analytic workspace, enabling you for the first time to use the transformation power on OLAP data loads.

Allowing you to load data from any source directly into the arena of high-end analytics of the OLAP option brings high-end analytics to the masses, allowing many people to access and work with OLAP data.

Executing the Load of the Cube



ORACLE®

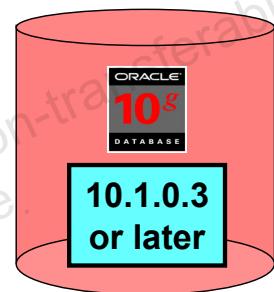
Copyright © 2006, Oracle. All rights reserved.

Executing the Load of the Cube

Execution of the mapping is similar to the method used to load relational mappings. You can set the Operating Mode parameter to “Set based” or “Row based.” The slower row-based method would be the equivalent of staging into local tables.

Database Version Required

- **Release 10.1.0.3 or later of Oracle Database 10g is required.**
- **Do not attempt OWB 10g Release 2 OLAP direct loading with Oracle Database 10g Release 10.1.0.2 or earlier.**



ORACLE

Copyright © 2006, Oracle. All rights reserved.

Database Version Required

The technology that enables direct deployment and loading of cubes was introduced with the 10.1.0.3 release of Oracle Database 10g.

Data Viewer for OLAP Objects

The Data Viewer
can pivot and
drill down
hierarchies.



It uses
BI Beans
components
dropped into
Warehouse
Builder.

		AMOUNT			COST		QUANTITY
		Channels Total	Indirect	Direct	Others	Channels Total	Channels Total
...	...	97,839,276	13,617,673	57,875,261	26,346,342	77,555,982	916,039
▼ Products Total							
▼ Hardware		20,643,006	2,549,724	12,248,462	5,844,819	16,214,712	15,353
▼ Portable PCs		15,009,043	1,543,130	9,044,989	4,420,924	11,277,822	9,589
Envoy Ambassador		15,009,043	1,543,130	9,044,989	4,420,924	11,277,822	9,589
► Desktop PCs		5,633,963	1,006,594	3,203,474	1,423,896	4,936,890	5,764
► Photo		17,961,866	2,742,668	10,573,992	4,645,206	13,384,536	95,509
► Electronics		14,595,163	2,224,272	8,241,518	4,129,373	12,384,191	115,973
► Software/Other		13,818,176	1,637,966	8,362,552	3,817,658	11,206,254	404,403
▼ Peripherals and Accessories		30,821,066	4,463,044	18,448,736	7,909,286	24,366,289	284,801
▼ Monitors		12,425,102	2,147,267	7,459,950	2,817,885	9,479,691	10,923
18" Flat Panel Graphics Monitor		5,235,931	1,090,473	3,017,889	1,127,569	4,106,682	4,913
17" LCD w/built-in HDTV Tuner		7,189,172	1,056,794	4,442,061	1,690,317	5,373,009	6,010
► Memory		4,903,595	369,342	3,334,360	1,199,893	4,020,914	35,208
► Printer Supplies		5,239,837	886,206	2,798,934	1,553,697	4,139,751	63,060
► Accessories		2,086,895	291,895	1,236,521	558,479	1,744,279	49,364
► Modems/Fax		2,185,351	313,084	1,150,326	721,940	1,771,917	43,716
CD-ROM		3,981,286	455,250	2,468,644	1,057,392	3,209,737	82,530

ORACLE

Copyright © 2006, Oracle. All rights reserved.

Data Viewer for OLAP Objects

The Data Viewer for dimensions and cubes is dimensionally aware. You get a fully functioning cross-tab that enables you to drill down hierarchies and pivot along dimensions.

You have *direct* analytic workspace access from within the OWB client.

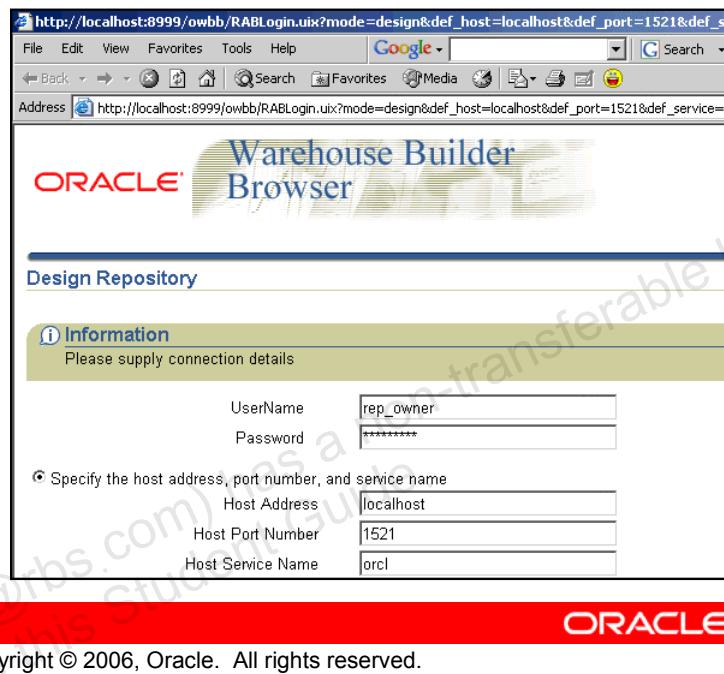
Data Viewer Validates Your Design and Loads

The Data Viewer makes it very easy for you to see whether your hierarchies drill down correctly and whether your loads are successful.

Monitoring Your OLAP Loads with the Repository Browser

Differences between OLAP and relational load monitoring:

None!
(from the user perspective)



Copyright © 2006, Oracle. All rights reserved.

Monitoring Your OLAP Loads with the Repository Browser

Use the Repository Browser to monitor your OLAP loads just as you would monitor your relational deployment and loading.

Summary

In this lesson, you should have learned that:

- **Warehouse Builder enables you to directly deploy OLAP cubes to the analytic workspace**
- **Warehouse Builder enables you to directly load OLAP cubes to the analytic workspace**
- **OLAP loading and execution monitoring is much the same as relational, from the user viewpoint**
- **You can use Warehouse Builder Data Viewer to view loaded cube data, including pivot and drill down**



Copyright © 2006, Oracle. All rights reserved.

Create the LOAD_SALES Mapping

Appendix D

ORACLE

Copyright © 2006, Oracle. All rights reserved.

Create the LOAD_SALES Mapping

Create the LOAD_SALES Mapping to Load the SALES Cube

1. From the Project Explorer, in the Design Center, expand CLASS_PROJECT > Databases > Oracle > SALES_WH. Right-click Mappings, and select New. The Create Mapping dialog box appears.
2. In the Create Mapping window, enter the following information:
Name: LOAD_SALES
Description: Mapping to load the Sales cube.
3. Click OK. The Mapping Editor is launched.
4. In the Explorer window in the Mapping Editor, ensure that the Available Objects tabbed page is displayed. Expand Oracle > XSALES > Tables.
5. Drag the ORDERS table onto the canvas. On the canvas, you see the ORDERS table operator.
6. On the canvas, right-click the header area of the ORDERS table operator. Select Open Details. The Table Editor window appears. In the Table Operator window, ensure that the General tabbed page is displayed. In the Name field, rename ORDERS to ORDERS_IN. Click OK.
7. In the Explorer panel in the Mapping Editor, ensure that the Available Objects tabbed page is displayed. Expand Oracle > XSALES > Tables.
8. Drag the ORDER_ITEMS table onto the canvas. On the canvas, right-click the header area of the ORDER_ITEMS table operator, and select Open Details. The Table Editor window appears.
9. In the Table Operator window, ensure that the General tabbed page is displayed. In the Name field, rename ORDER_ITEMS to ORDER_ITEMS_IN. Click OK. Now, on the canvas, you see two operators, ORDERS_IN and ORDER_ITEMS_IN.
10. From the Explorer panel, drag the ADDRESSES table onto the canvas (the ADDRESSES table from the XSALES schema). On the canvas, right-click the header area of the ADDRESSES table operator. Select Open Details. The Table Editor window appears.
11. In the Table Operator window, ensure that the General tabbed page is displayed. In the Name field, rename ADDRESSES to ADDRESSES_IN. Click OK.

Create the LOAD_SALES Mapping (continued)

12. From the Explorer panel, drag the PROMOTIONS table onto the canvas (the ADDRESSES table from the XSALES schema).
13. On the canvas, right-click the header area of the PROMOTIONS table operator. Select Open Details. The Table Editor window appears.
14. In the Table Operator window, ensure that the General tabbed page is displayed. In the Name field, rename PROMOTIONS to PROMOTIONS_IN. Click OK.
15. In the Explorer panel in the Mapping Editor, click the Available Objects tab. Expand Oracle > SALES_WH > Cubes. Drag the SALES cube onto the canvas.
16. On the canvas, right-click the header area of the SALES cube operator. Select Open Details. The Table Editor window appears.
17. In the Table Operator window, ensure that the General tabbed page is displayed. In the Name field, rename SALES to SALES_OUT. Click OK.

On the canvas, you see five operators, four table operators and one cube operator. You continue designing the cube mapping.

Add a Joiner

18. From the Palette window, drag the joiner operator onto the canvas. The joiner operator is named as JOINER. By default, the joiner creates three groups for you, two Input groups and one Output group. The Input groups are named as INGRP1 and INGRP2, and the Output group is named as OUTGRP1.

Add Two Additional Input Groups to the JOINER Operator

19. On the canvas, right-click the header area of the JOINER operator. Select Open Details. The Joiner Editor window appears.
20. In the Joiner Editor, click the Groups tab. You see a total of three groups, two Input groups and one Output group.
21. Click Add. In the Group column, INGRP3 is added, and in the Direction column, you see INGRP.
22. Add another group. Click Add. In the Group column, INGRP4 is added, and in the Direction column, you see INGRP.

Now you add connection lines from the various table operators that you added onto the canvas to the JOINER operator.

Create the LOAD_SALES Mapping (continued)

Draw Connection Lines from Attributes in INOUTGRP1 of the ORDERS_IN Table Operator to INGRP1 of the JOINER Operator

23. Draw connection lines from attributes in INOUTGRP1 of the ORDERS_IN table operator to INGRP1 of the JOINER operator as:

ORDERS_IN.INOUTGRP1	JOINER
ID	INGRP1
CUSTOMER_ID	INGRP1
CHANNEL	INGRP1
PROMOTION_ID	INGRP1
ORDER_FINISHED	INGRP1

Draw Connection Lines from Attributes in INOUTGRP1 of the ORDER_ITEMS_IN Table Operator to INGRP2 of the JOINER Operator

24. Add connection lines from attributes in INOUTGRP1 of the ORDER_ITEMS_IN table operator to INGRP2 of the JOINER operator as:

ORDER_ITEMS_IN.INOUTGRP1	JOINER
ORDER_ID	INGRP2
PRODUCT_ID	INGRP2
AMOUNT	INGRP2
COST	INGRP2
QUANTITY	INGRP2

Create the LOAD_SALES Mapping (continued)

Draw Connection Lines from Attributes in INOUTGRP1 of the ADDRESSES_IN Table Operator to INGRP3 of the JOINER Operator

25. Draw a connection line from attributes in INOUTGRP1 of the ADDRESSES_IN table operator to INGRP3 of the JOINER operator as:

ADDRESSES_IN.INOUTGRP1	JOINER
CUSTOMER_ID	INGRP3
CITY_ID	INGRP3

Add a Connection Line from INOUTGRP1 of the PROMOTIONS_IN Table Operator to INGRP4 of the JOINER Operator

PROMOTIONS_IN.INOUTGRP1	JOINER
ID	INGRP4
SUBCATEGORY_ID	INGRP4

Now you have connected the source operators with the joiner operator. Now you specify the join condition for the JOINER joiner operator.

Define the Join Condition

26. On the canvas in the Mapping Editor, click the header area of the JOINER operator. On the left is the Property Inspector window. In the Property Inspector window, click the field to the right of Join Condition. Click ... (three dotted button). The Expression Builder window appears.
27. In the Expression Builder window, you build your expression instead of having to type the expression. On the left, ensure that the Inputs tabbed page is displayed. Expand INGRP1. Double-click ID. On the right, you see that INGRP1.ID is displayed. Click = (equal to) in the section that is below the join condition. The expression now appears as follows:
INGRP1.ID =
28. On the left of the Expression Builder, ensure that the Inputs tabbed page is displayed. Expand INGRP2. Double-click ORDER_ID. The expression now looks like this:
INGRP1.ID = INGRP2.ORDER_ID
29. Click AND.
30. With INGRP1 expanded, double-click CUSTOMER_ID.
31. Click = (equal to). Expand INGRP3, and double-click CUSTOMER_ID. The expression looks like this: INGRP1.ID = INGRP2.ORDER_ID AND INGRP1.CUSTOMER_ID = INGRP3.CUSTOMER_ID
32. Click AND.
33. With INGRP1 expanded, double-click PROMOTION_ID.
34. Click = (equal to). Expand INGRP4, and double-click ID.

Create the LOAD_SALES Mapping (continued)

35. You have built the entire expression, INGRP1.ID = INGRP2.ORDER_ID and INGRP1.CUSTOMER_ID = INGRP3.CUSTOMER_ID and INGRP1.PROMOTION_ID = INGRP4.ID.

You now add a Key Lookup operator to the LOAD_SALES cube mapping.

Add a Key Lookup Operator

36. In the Mapping Editor, from the Palette window, drag the key-lookup operator onto the canvas. The Key Lookup Wizard is launched. Click Next on the Welcome page.
37. On the General page, enter CHANNELS_LKP as the name for the Key-Lookup operator. Click Next.
38. Examine the Groups page; it has two groups, INGRP1 and OUTGRP1. Click Next.
39. On the Input connection page, in the Available Attributes section, expand JOINER > OUTGRP1. Select CHANNEL and move it to the Mapped Attributes list. Click Next.
40. On the Lookup page, from the “Select the object which has the lookup result” drop-down box, select CHANNELS. Below, in the Lookup condition section, click in the first row in the lookup table key column. From the drop-down list, select NAME. In the Input Attribute column, select CHANNEL. Click Next.
41. On the No-match Rows page, select “Return a row with the following default values.” This page enables you to specify how you want to handle a row that does not find a match. Click Next.
42. Review the Summary details, and click Finish.

Create the LOAD_SALES Mapping (continued)

Add an Expression Operator

43. From the Palette window in the Mapping Editor, drag the expression operator onto the canvas.
44. Right-click the header area of the expression operator, and select Open Details.
45. In the Expression Editor, ensure that the General tabbed page is displayed. Rename the expression operator to TO_NUM_EXPR.
46. Drag a connection line from the ORDER_FINISHED attribute in OUTGRP1 of the JOINER operator to INGRP1 of the TO_NUM_EXPR expression operator.
47. Right-click the header area of the TO_NUM_EXPR expression operator, and select Open Details.
48. In the Expression Editor, click the Output Attributes tab. Click Add. In the Attribute column, rename OUTPUT1 to FINISH_MONTH. Ensure that the data type is NUMBER. Click OK.
49. On the canvas, click the FINISH_MONTH attribute in OUTGRP1 of the TO_NUM_EXPR expression operator.
50. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button).
51. In the Expression Editor window, on the left, click the Transformations tab. Expand Transformation > Public > Pre-Defined > Conversion. Scroll down to TO_NUMBER. Double-click TO_NUMBER (UNSPECIFIED, VARCHAR2, VARCHAR2).
52. In the Expression section, the TO_NUMBER function, along with parameters, appears. You now replace the parameters with values.
53. Select and highlight UNSPECIFIED. Replace UNSPECIFIED with the following.
54. On the left of the Expression Editor, ensure that the Transformation tabbed page is displayed. Expand Transformation > Public > Pre-Defined > Conversion. Scroll down to TO_CHAR. Double-click TO_CHAR(UNSPECIFIED, VARCHAR2, VARCHAR2). The complete expression that you see looks like this: TO_NUMBER (TO_CHAR (UNSPECIFIED, VARCHAR2, VARCHAR2) , VARCHAR2, VARCHAR2)
55. Select and highlight UNSPECIFIED. Replace UNSPECIFIED with the following.
56. On the left of the Expression Editor, ensure that the Inputs tabbed page is displayed. Expand INGRP1. Double-click ORDER_FINISHED. The complete expression that you see looks like this: TO_NUMBER (TO_CHAR (INGRP1.ORDER_FINISHED, VARCHAR2, VARCHAR2) , VARCHAR2, VARCHAR2)
57. Now complete the expression to look like this:
TO_NUMBER (TO_CHAR (INGRP1.ORDER_FINISHED, 'YYYYMM') , '999999')
58. Click Validate. If the validation is successful, click OK.

Create the LOAD_SALES Mapping (continued)

Add an Aggregator Operator

59. In the Mapping Editor, from the Palette window, drag the AGGREGATOR operator onto the canvas.

Draw Connection Lines from Attributes in OUTGRP1 of the JOINER Operator to INGRP1 of the AGGREGATOR Operator

60. Drag a connection line from attributes in OUTGRP1 of JOINER to INGRP1 of the AGGREGATOR operator as:

JOINER. OUTGRP1	AGGREGATOR
AMOUNT	INGRP1
COST	INGRP1
QUANTITY	INGRP1
CITY_ID	INGRP1
PRODUCT_ID	INGRP1
SUBCATEGORY_ID	

Draw Connection Lines from the Attribute in OUTGRP1 of the CHANNELS_LKP Key Lookup Operator to INGRP1 of the AGGREGATOR Operator

61. Drag a connection line from the attribute in OUTGRP1 of CHANNELS_LKP to INGRP1 of the AGGREGATOR operator as:

CHANNELS_LKP. OUTGRP1	AGGREGATOR
ID	INGRP1

Draw a Connection Line from the Attribute in OUTGRP1 of the TO_NUM_EXPR Expression Operator to INGRP1 of the AGGREGATOR Operator

62. Drag a connection line from FINISH_MONTH of the TO_NUM_EXPR expression operator to INGRP1 of the AGGREGATOR operator as:

TO_NUM_EXPR. OUTGRP1	AGGREGATOR
FINISH_MONTH	INGRP1

Create the LOAD_SALES Mapping (continued)

Add attributes to the OUTGRP1 group of the AGGREGATOR operator.

63. On the canvas, in the Mapping editor, right-click the header area of the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
64. Click Add. Rename OUTPUT1 to AMOUNT, and specify the other details as:

Data Type: NUMBER

Precision: 10

Scale: 2

65. Click OK.

66. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select AMOUNT. On the left, in the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:

Function: SUM

All/Distinct: All

Attribute: AMOUNT

67. Click OK.

68. On the canvas, in the Mapping editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.

69. Click Add. Rename OUTPUT1 to COST. Specify these details:

Data Type: NUMBER

Precision: 10

Scale: 2

70. In the Mapping Editor, click the header area of the AGGREGATOR operator.

71. Expand OUTGRP1, and select COST. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:

Function: SUM

All/Distinct: All

Attribute: COST

72. Click OK.

Create the LOAD_SALES Mapping (continued)

73. On the canvas, in the Mapping editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
74. Click Add. Rename OUTPUT1 to QUANTITY, and specify the following:
Data Type: NUMBER
75. Click OK.
76. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select QUANTITY. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:
Function: SUM
All/Distinct: All
Attribute: QUANTITY
77. Click OK.
78. On the canvas, in the Mapping Editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
79. Click Add. Rename OUTPUT1 to CHANNEL_SOURCE_ID, and specify the following:
Data Type: NUMBER
80. Click OK.
81. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select CHANNEL_SOURCE_ID. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:
Function: NONE
Attribute: ID
82. Click OK.
83. On the canvas, in the Mapping editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
84. Click Add. Rename OUTPUT1 to CITY_ID, and specify the following:
Data Type: NUMBER
85. Click OK.

Create the LOAD_SALES Mapping (continued)

86. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select CITY_ID. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:
Function: NONE
Attribute: CITY_ID
87. Click OK.
88. On the canvas, in the Mapping editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
89. Click Add. Rename OUTPUT1 to PRODUCT_ID. Specify the following:
Data Type: NUMBER
90. Click OK.
91. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select PRODUCT_ID. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:
Function: NONE
Attribute: PRODUCT_ID
92. Click OK.

Create the LOAD_SALES Mapping (continued)

93. On the canvas, in the Mapping Editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
94. Click Add. Rename OUTPUT1 to SUBCATEGORY_ID, and specify the following:
Data Type: NUMBER
95. Click OK.
96. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select SUBCATEGORY_ID. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:
Function: NONE
Attribute: SUBCATEGORY_ID
97. Click OK.
98. On the canvas, in the Mapping Editor, right-click the AGGREGATOR operator. Select Open Details. Click the Output Attributes tab.
99. Click Add. Rename OUTPUT1 to FINISH_MONTH, and specify the following:
Data Type: NUMBER
100. Click OK.
101. In the Mapping Editor, click the header area of the AGGREGATOR operator. Expand OUTGRP1, and select FINISH_MONTH. In the Property Inspector window, click on the right of the Expression property. Click ... (three dotted button). The Expression Editor is launched. In the Expression window, from the drop-down lists, select the following:
Function: NONE
Attribute: FINISH_MONTH
102. Click OK.

Create the LOAD_SALES Mapping (continued)

Draw Connection Lines from Attributes in the OUTGRP1 Group of the AGGREGATOR Operator to Attributes in the SALES Group of the SALES_OUT Cube Operator

103. Drag connection lines from attributes in OUTGRP1 of the AGGREGATOR operator in the SALES group of the SALES_OUT cube operator as:

AGGREGATOR.OUTGRP1	SALES_OUT
AMOUNT	SALES.AMOUNT
COST	SALES.AMOUNT
QUANTITY	SALES.QUANTITY
CHANNEL_SOURCE_ID	SALES.CHANNEL_SOURCE_ID
SUBCATEGORY_ID	SALES.PROMOTIONS_SOURCE_ID
PRODUCT_ID	SALES.PRODUCTS_SOURCE_ID
CITY_ID	SALES.CUSTOMERS_SOURCE_ID
FINISH_MONTH	SALES.TIMES_CAL_MONTH_CODE

104. In the Mapping Editor, click the header area of the SALES_OUT cube operator.
105. In the Property Inspector window, navigate to the Solve the Cube property and from the drop-down list, select YES. When you specify YES, cube data is precomputed for the levels specified in the definition of cube.
106. In the Mapping Editor, from the Mapping menu, select Validate.
107. From the Mapping menu, select Generate. This generates the code required to implement the design. Close the Mapping Editor.
108. In the Design Center, from the Design menu, select Save All, or click the Save All toolbar icon.
109. To save time, the LOAD_CHANNELS, LOAD_CUSTOMERS, and LOAD_PROMOTIONS mappings have been created for you. TIMES_MAP got created for you when you created the TIMES dimension by using Time Wizard.
110. In the Mapping Editor, from the Mapping menu, select Generate. Warehouse Builder generates the required code to implement the design.
111. Close the Mapping Editor. In the Design Center, click the Save All toolbar icon. Exit OWB.

Create the LOAD_SALES Mapping (continued)

112. The all.tcl script calls other scripts to create a number of repository objects. This completes the setup for you to save time.
113. When the script finishes running, at the OMB+> prompt, type exit and press [Enter] to exit OMB Plus.

Log In to Warehouse Builder Design Client

114. From the Windows Start menu, select Programs > Oracle - OWBHome10g R2 > Warehouse Builder > Design Center to invoke Warehouse Builder.

vivek kamra (vivek.kamra@rbs.com) has a non-transferable license to
use this Student Guide.