PowerCenter 8.x
Level I Developer
Lab Guide

Version – L1D_20081124G

PowerCenter 8.x Level I Developer Lab Guide

Version 04

November 2008

Table of Contents

Lab


INFORMATICA
GLOBAL
EDUCATION SERVICES

Lab 2-1: Creating Source Definitions

Scenario:

- You will create some Source definitions for use in later work.

Goals:

- Use wizards to import Source definitions from a flat file and a relational database table
- Preview the data in the Sources

Duration:

10 minutes

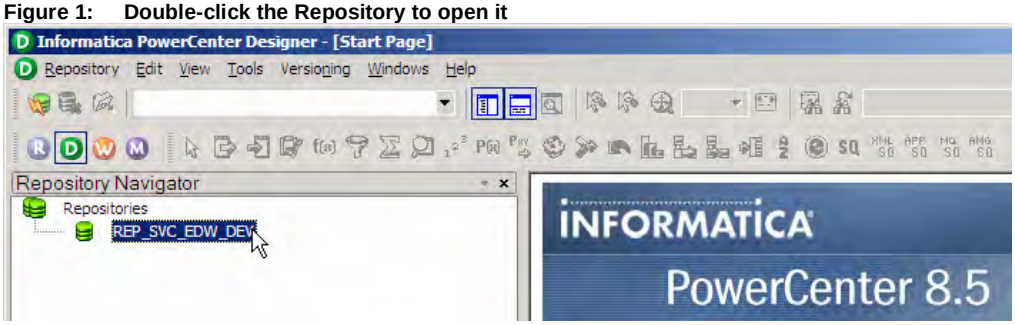
Instructions

> **Note:** Throughout this and later exercises, *xx* will refer to the student number assigned to you by your Instructor or the machine you are working on. For example, if you are Student05, then ~Developer*xx* refers to the folder ~Developer05.

Step 1. Start PowerCenter Designer

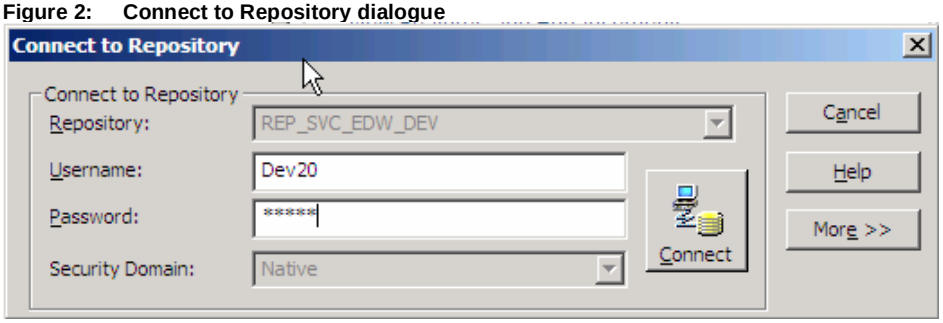
1) On your desktop, double-click the PowerCenter Designer icon ( ) to start it.
2) In the Repository Navigator, double-click REP_SVC_EDW_DEV.

Figure 1: Double-click the Repository to open it



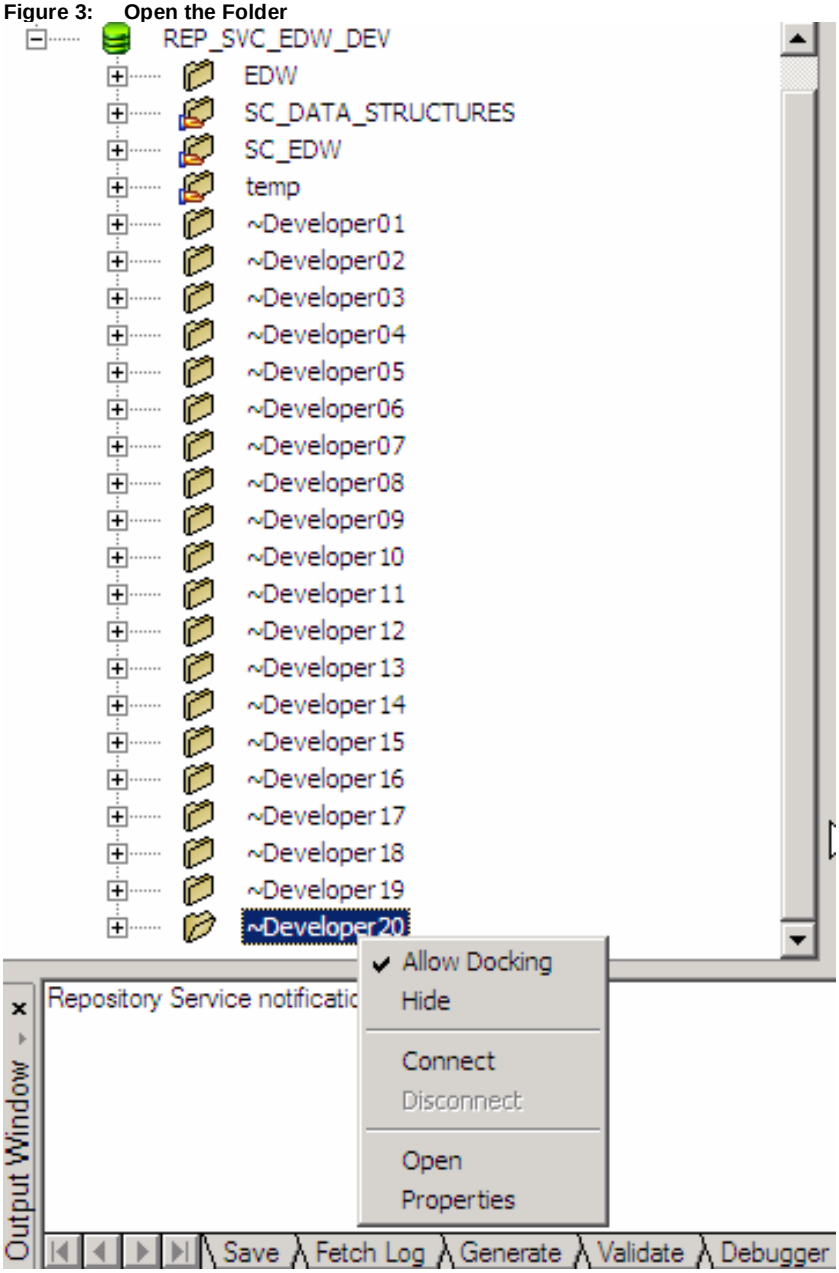
3) In the "Connect to Repository" dialogue:
   a) For Username, enter Dev*xx* (*xx* is the number assigned by your instructor).
   b) For Password, enter Dev*xx*.
   c) Click **Connect**.

Figure 2: Connect to Repository dialogue



4) Right-click the folder labeled ~Developer*xx* and select **Open**.

Figure 3: Open the Folder



Note: In future instructions this may be referred to as "your folder."

Step 2. Create a Flat File Source

1) From the menu, select **Tools → Source Analyzer**.
2) From the menu, select **Sources → Import from File**.
3) In the Open Flat File dialogue, select **customer_central.dat** and click **OK**.
   a) The file is located in C:\infa-shared\Srcfiles
4) In Step 1 of the Flat File Import Wizard:
   a) Make sure that the **Delimited** radio button is selected.

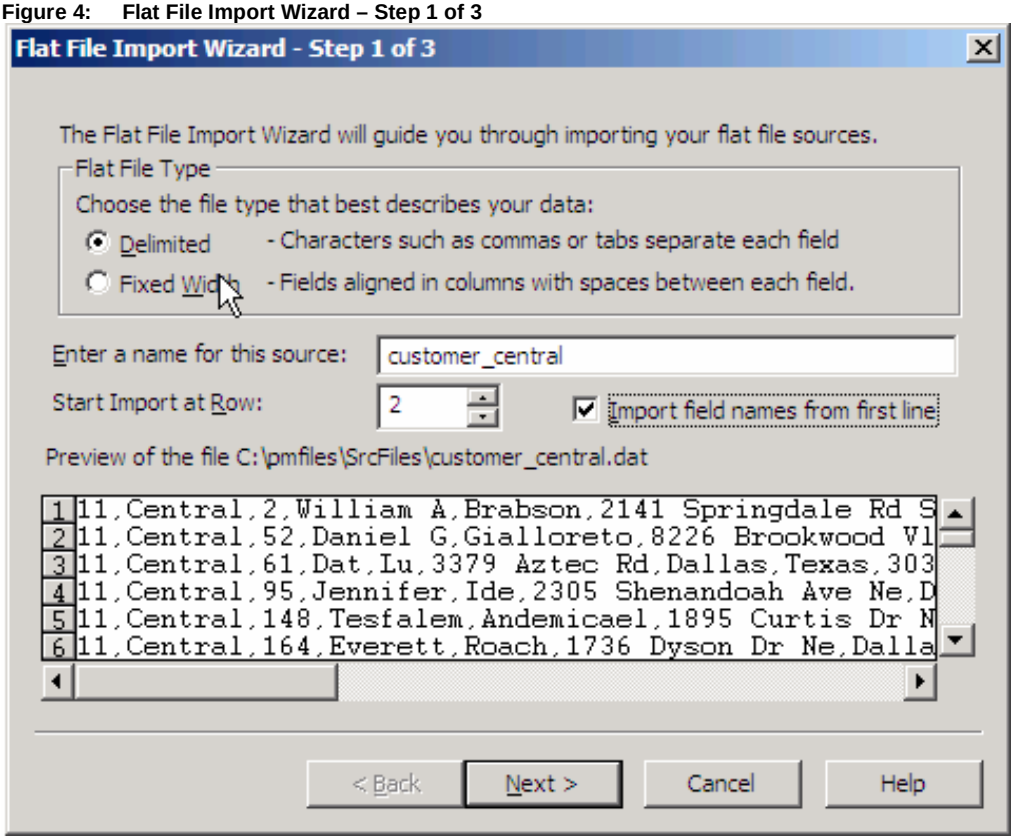
Note: This is because you are importing a comma-delimited file. You will select the field delimiter on the next screen. Note that PowerCenter can also import files with fixed field widths.

   b) Check **Import field names from first line**.
      (i) Note that "Start Import at Row" automatically changes to "2".

Note: When a file has column names, as this file does, PowerCenter can import those as field names.

   c) Click **Next**.

Figure 4: Flat File Import Wizard – Step 1 of 3



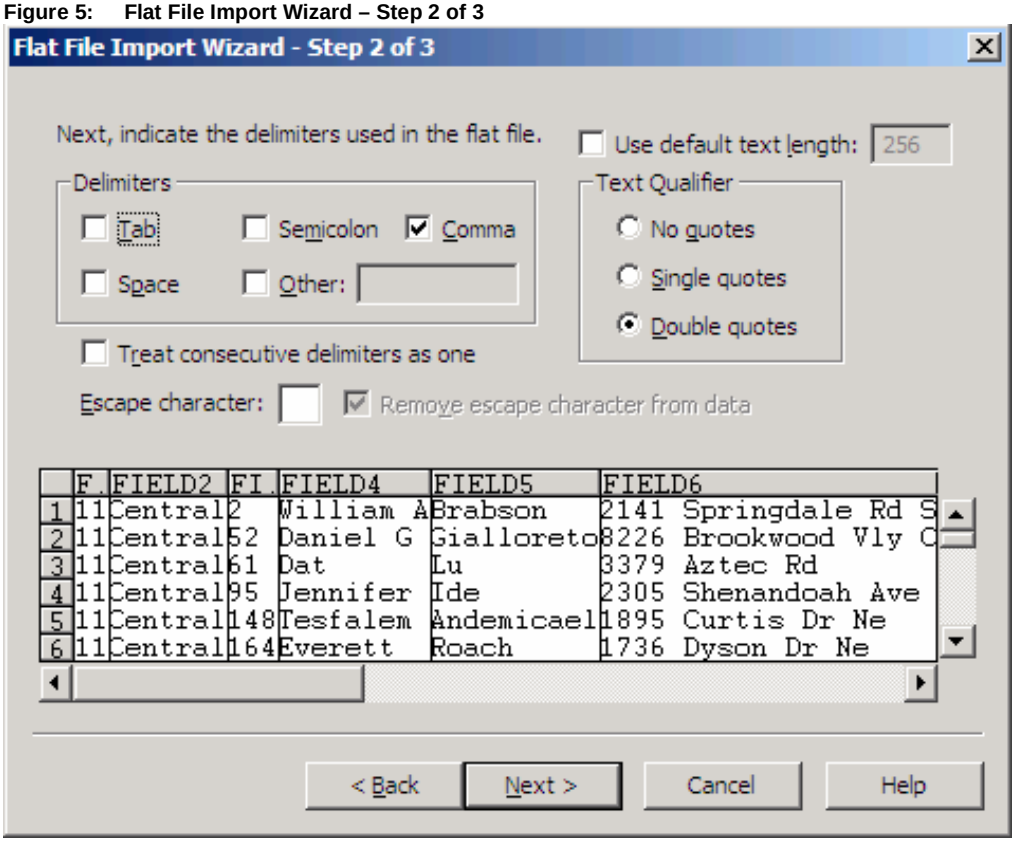
5) In Step 2 of the Flat File Import Wizard:

a) In the Delimiters section, make sure that **Comma** is checked.

Note: While a number of standard delimiters are listed, you can define any character or set of characters as the delimiter using the "Other" checkbox.

b) Accept the defaults for other values and click **Next**.

Figure 5: Flat File Import Wizard – Step 2 of 3



Note: This step sets up the fields in general. You will have the opportunity to adjust individual fields in Step 3.

Note: "Use default text length" – check this to set a standard length for all fields with a text data type. Leave it unchecked and PowerCenter derives the text field length from the actual length of the data in the file.

Note: "Escape Character" is the character used in your file format if the delimiter character may appear in a field. Consult documentation to learn more.

6) In Step 3 of the Flat File Import Wizard:

a) Use the scrollbar to move to the right and select the City field.

b) Change the length/precision of the City field to 50

c) Change the length/precision of the State field to 15

Note: These lengths come from data analysis.

d) Select the ZIP field.

Note: You will adjust this field because you know in the future you will import addresses with Canadian postal codes, which contain alphanumeric characters.

e) Change the field type to **Text**.

f) Change the Length/Precision to **6**.

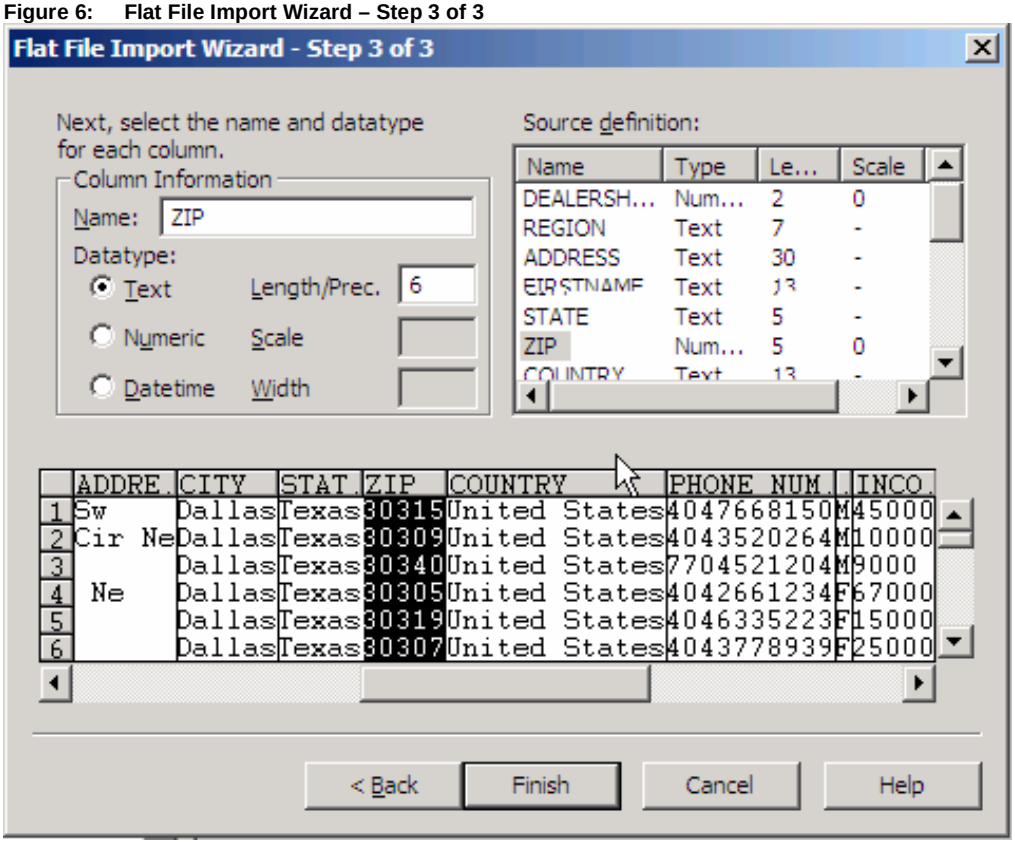
g) Scroll to the far right and select the field DATE.

h) Change the field type to Datetime.

Note: You will adjust this field because you know that in the target database the data will be stored as a Date. You could perform the conversion using PowerCenter's "To-Date" function, but it is simpler to use the implied conversion functionality of the Source definition.

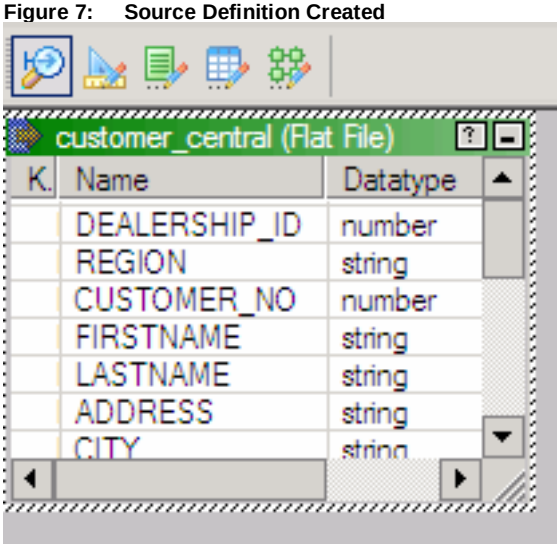
i) Click **Finish**.

Figure 6: Flat File Import Wizard – Step 3 of 3



7) The Source definition will appear in the Source Analyzer workspace.

Figure 7: Source Definition Created



8) Edit the Source definition.

Note: You may use this same Source definition to import data from multiple flat files. You will change the name to refer generically to customers rather than the specific customer data file.

a) Double-click the green header bar at the top of the Source definition.

(i) This opens the definition for editing.

b) In the Edit Tables dialogue, click **Rename**.

c) For Table Name, enter **Customers**.

d) Click **OK**

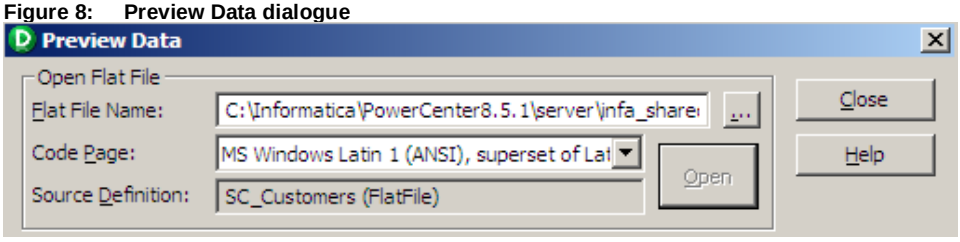
e) Click **OK**.

9) To verify that the Source imported correctly, you will now preview the contents of the flat file.

a) Right-click the header bar of the Source definition and select **Preview data**.

b) In the Preview Data dialogue:

Figure 8: Preview Data dialogue



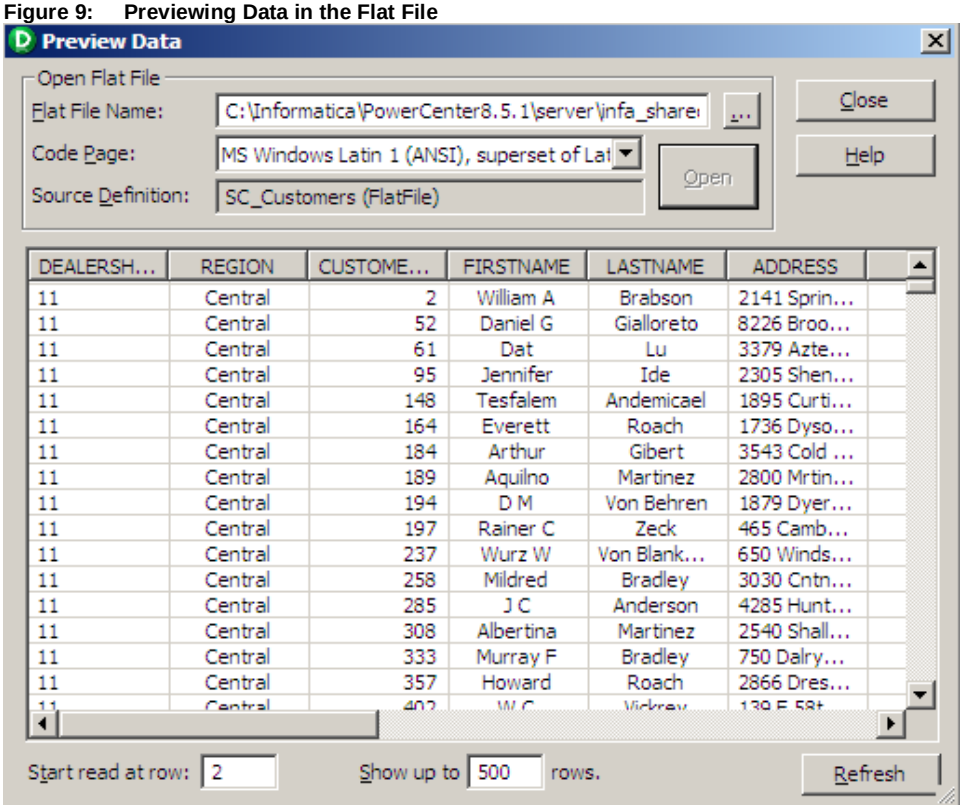
(i) Click the button labeled with three dots (…).

(ii) In the Open Flat File dialogue, select **customer_central.dat** and click **Open**.

(iii) Click **Open**.

(iv) The Preview Data dialogue will display data from the flat file.

Figure 9: Previewing Data in the Flat File

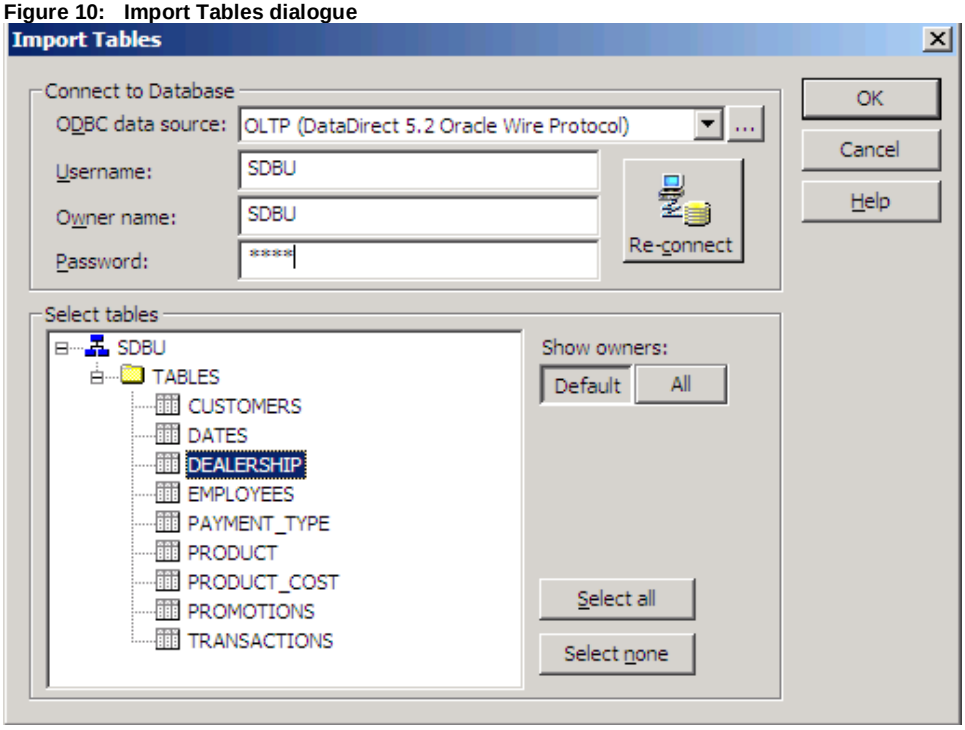


(v) Click **Close**.

Step 3. Create a Relational Database Table Source

1) Right-click on the Source Analyzer workspace and select **Clear All**.
2) Select **Sources → Import from Database**.
3) In the Import Tables dialogue:
   a) For ODBC data source, select **OLTP** (**DataDirect 5.2 Oracle Wire Protocol**).
   b) For Username, enter **SDBU**.
      (i) Owner Name will automatically populate with SDBU
   c) For Password, enter **SDBU**.
   d) Click **Connect**.

Figure 10: Import Tables dialogue



e) In the Select tables pane, click the plus sign (+) beside **SDBU** to expand it.

f) Expand **TABLES** similarly.

g) Select **DEALERSHIP**.

h) Click **OK**.

i) The source definition DEALERSHIP (Oracle) appears in the Source Analyzer workspace.

4) As before, verify the Source by previewing the data.

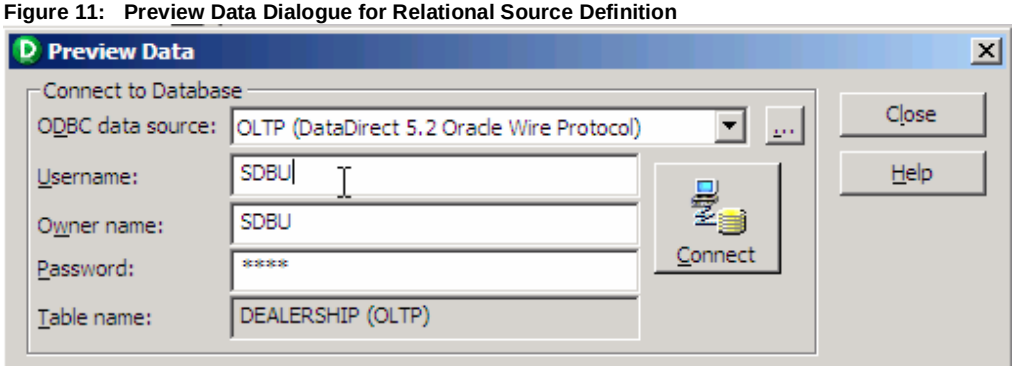
a) Right-click the Source and select **Preview Data**.

b) In the Preview Data dialogue:

(i) Select ODBC data source **OLTP (DataDirect 5.2 Oracle Wire Protocol)**.

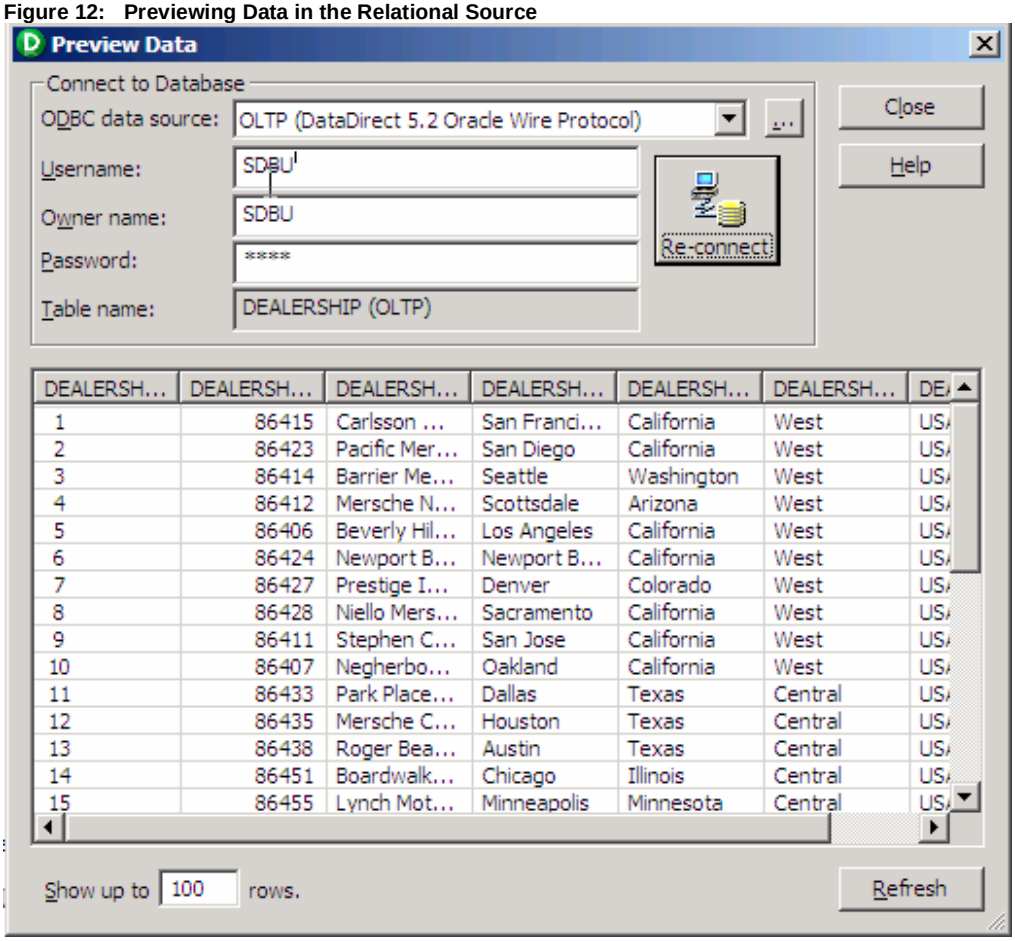
(ii) For Username, Owner name, and Password, enter **SDBU**. (Owner name should populate automatically.)

Figure 11: Preview Data Dialogue for Relational Source Definition



(iii) Click **Connect**.

Figure 12: Previewing Data in the Relational Source



(iv) Click **Close**.

Step 4. Save Your Work

1) Type **Ctrl+S** to save your work.

Note: Always save your work before closing the application or moving on to another task. There is no automatic save in PowerCenter.

Note: You can also save by selecting **Repository→Save** from the menu.

Lab 2-2: Creating Target Definitions

Scenario:

- You will create a relational Target to receive data from the flatfile Source you created in Lab 1

Goals:

- Create a Target definition from scratch
- Create a Target definition from a Source definition and change the Target type

Duration:

10 minutes

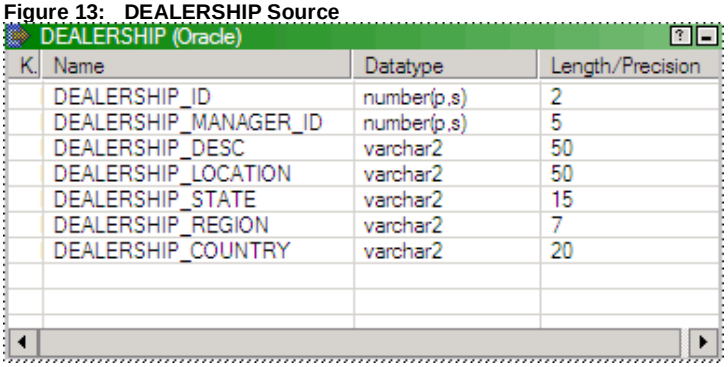
Instructions

Step 1. Define a Target

1) Determine what columns will be required

   a) In PowerCenter Designer, drag the DEALERSHIP Source onto the Source Analyzer workspace and examine it to determine what columns a target based on this source will require.

Figure 13: DEALERSHIP Source



DEALERSHIP (Oracle)

K.	Name	Datatype	Length/Precision
	DEALERSHIP_ID	number(p,s)	2
	DEALERSHIP_MANAGER_ID	number(p,s)	5
	DEALERSHIP_DESC	varchar2	50
	DEALERSHIP_LOCATION	varchar2	50
	DEALERSHIP_STATE	varchar2	15
	DEALERSHIP_REGION	varchar2	7
	DEALERSHIP_COUNTRY	varchar2	20

2) Create a Target definition.

   a) In PowerCenter Designer, select the Target Designer tool ().

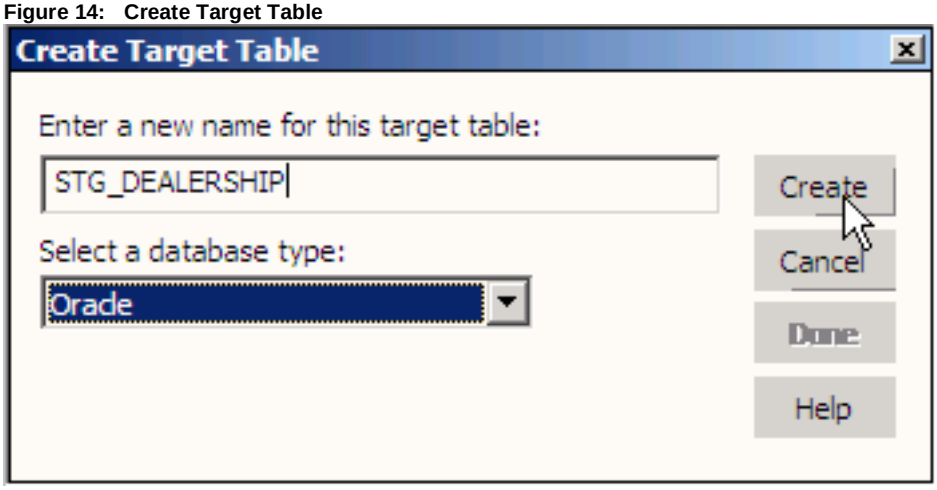
   b) From the Designer menu, select **Targets→Create**.

c) In the Create Target Table dialogue:

(i) Enter the name **STG_DEALERSHIP**

(ii) Select the database type **Oracle**

Figure 14: Create Target Table

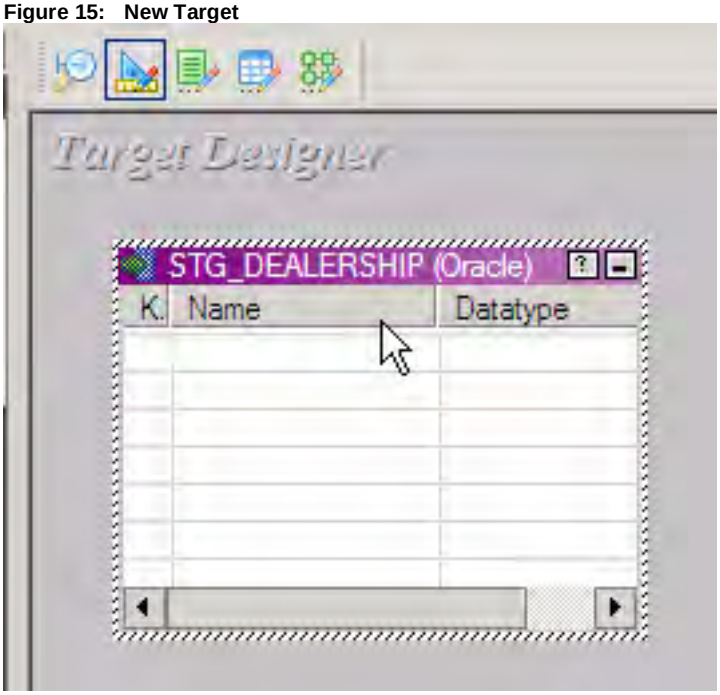


(iii) Click **Create.**

(iv) Click **Done.**

d) The Target definition appears in the Target Designer workspace, with no columns.

Figure 15: New Target

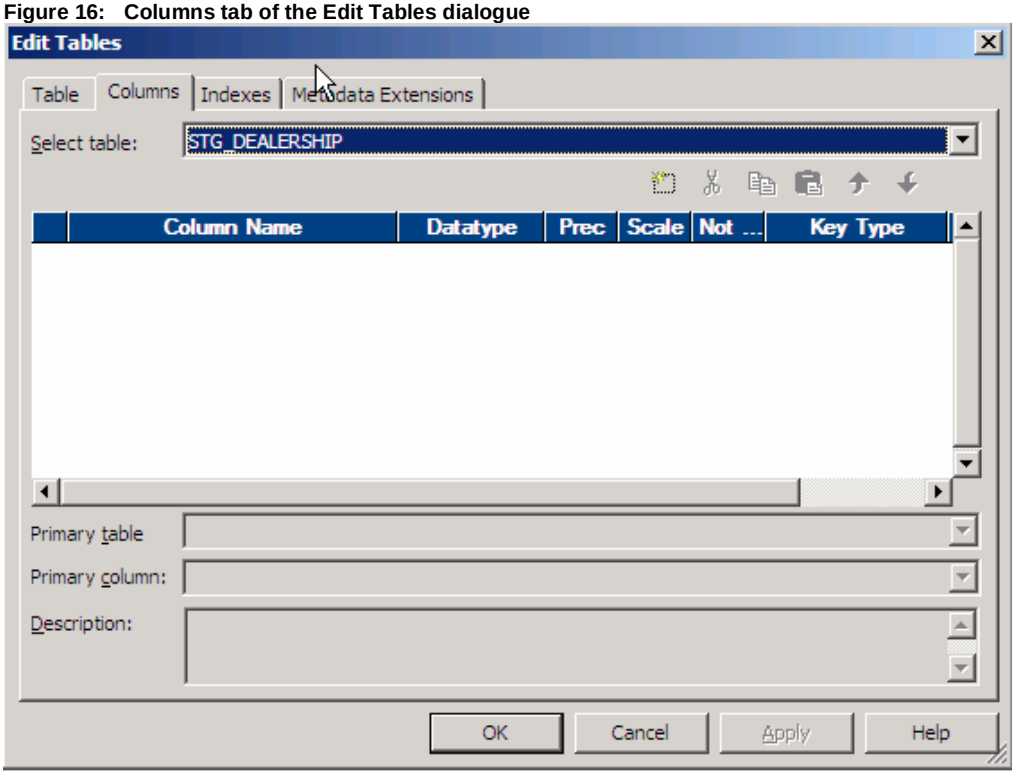


3) Add a column to the Target definition.

a) Double-click the header of the STG_DEALERHIP Target definition to open the Edit Tables dialogue.

b) Select the **Columns** tab.

Figure 16: Columns tab of the Edit Tables dialogue



c) Click the New Column button ( ) to create a new column.

(i) For the Name, enter **DEALERSHIP_ID**

(ii) Set the Datatype to **number(p,s)**

(iii) For the Precision, enter **3**

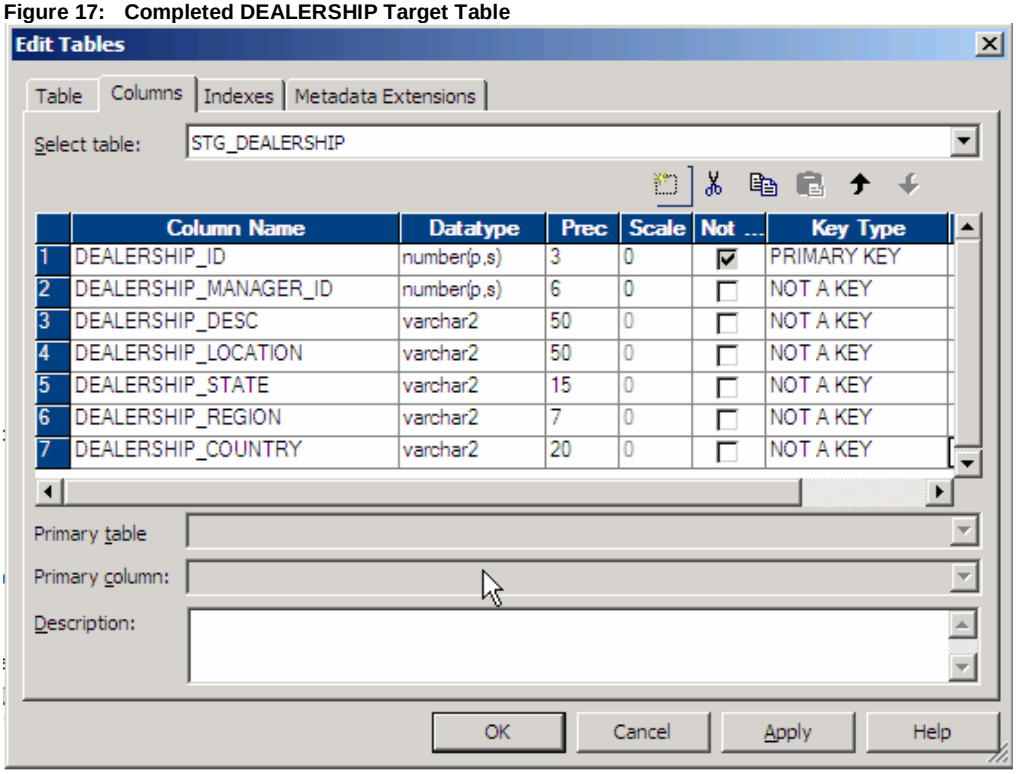
(iv) For Scale, leave the default **0**

(v) **DEALERSHIP_ID** is the unique identifier of a dealership. Therefore, select **Primary Key** in the Key Type column.

d) Repeat this process for the remaining six columns.

(i) Accept the default (Not a Key) for Key Type.

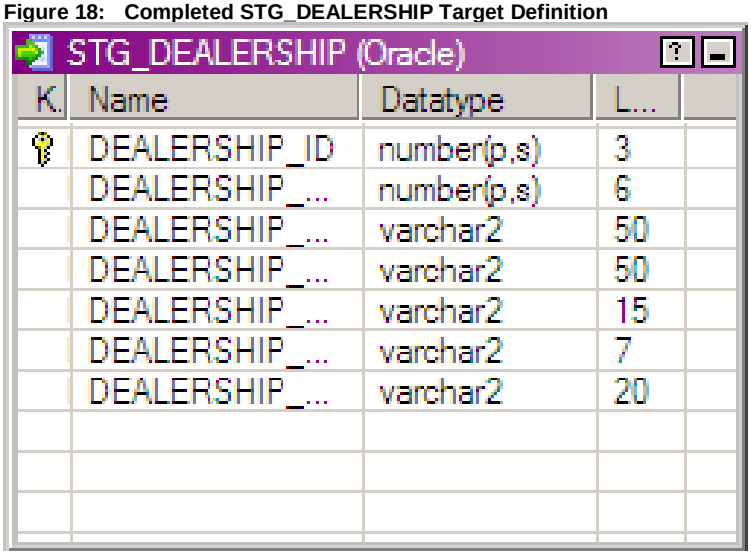
Figure 17: Completed DEALERSHIP Target Table



e) When finished, click **Apply** then **OK**.

f) The Target definition should look like this:

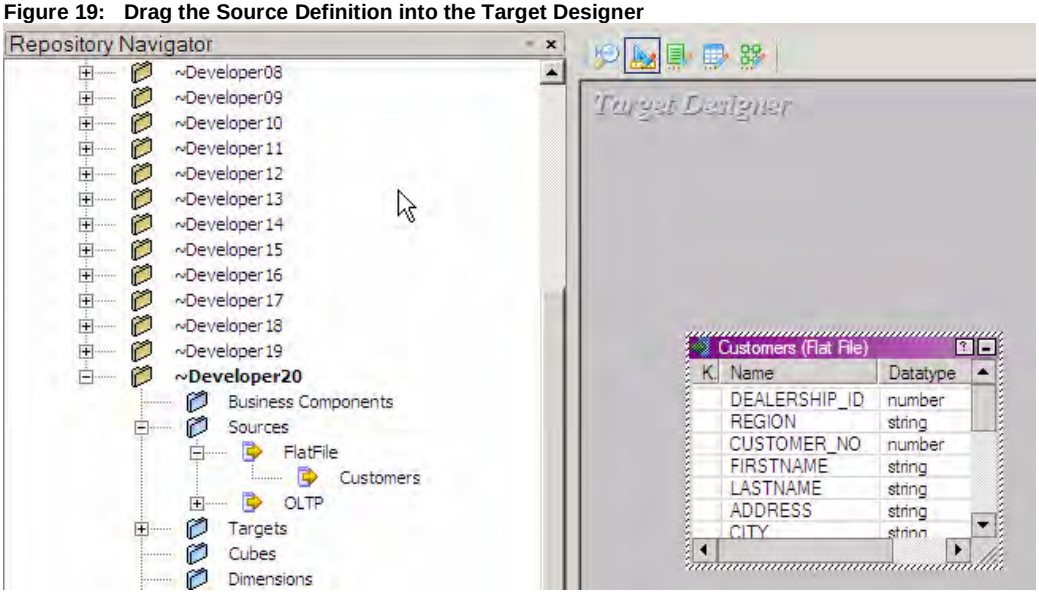
Figure 18: Completed STG_DEALERSHIP Target Definition



Step 2. Create a Target Definition from a Source

1) In the Repository Navigator, locate the **Customers** flat file Source definition you created in the previous lab and drag it into the Target Designer workspace.

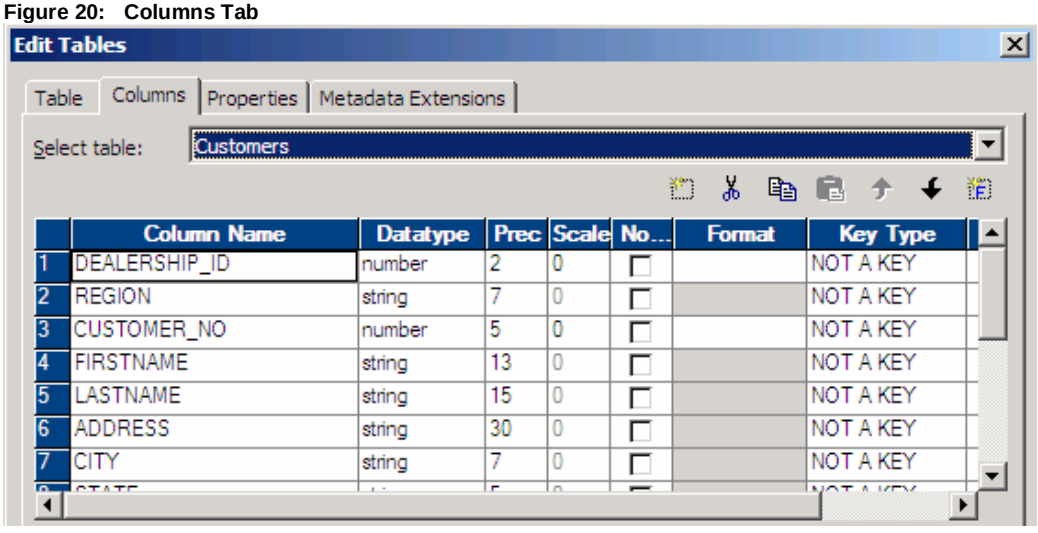
Figure 19: Drag the Source Definition into the Target Designer



2) Double-click the header of the Customers Target definition to open the Edit Tables dialogue and select the Columns tab.

   a) Note that the Datatypes are "number" and "string," as is standard for flat file definitions.

Figure 20: Columns Tab



3) Select the Table tab.

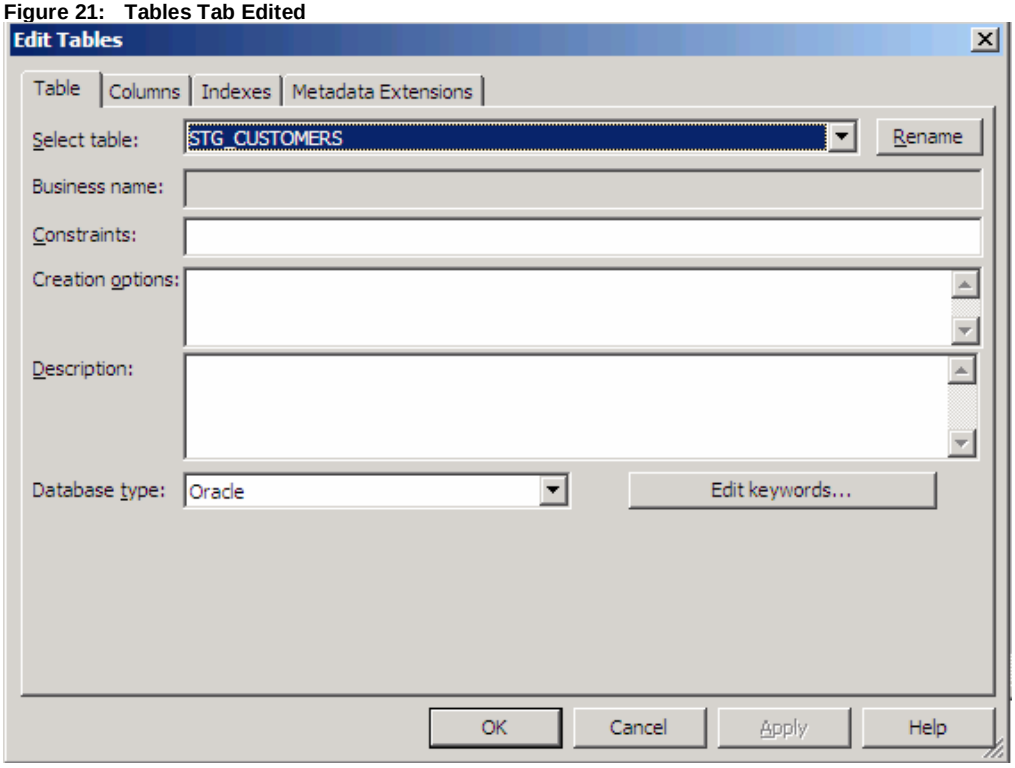
4) Click the Rename button and change the target name to **STG_CUSTOMERS**.

5) Change the "Database type" dropdown to **Oracle**.

6) Click **Apply**.

   a) The Tables tab should now look like this:

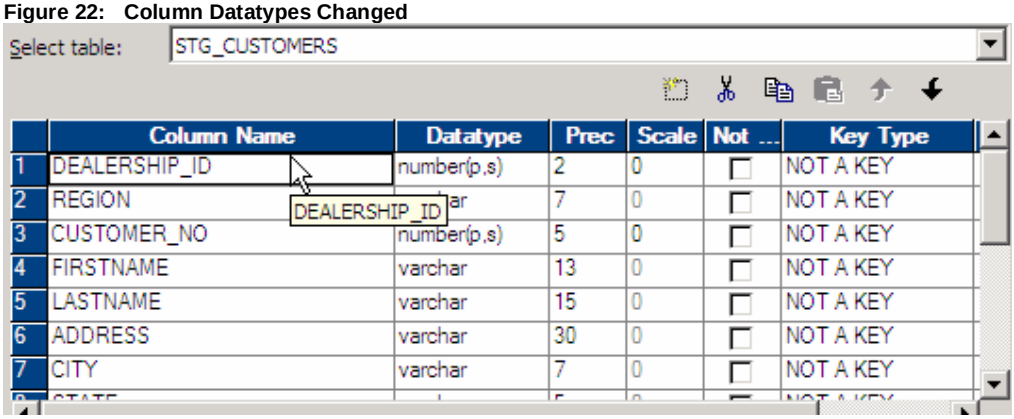
Figure 21: Tables Tab Edited



7) Return to the Columns tab.

   a) Note that the "number" and "string" datatypes have changed to the "number(p,s)" and "varchar" types appropriate to Oracle.

Figure 22: Column Datatypes Changed



	Column Name	Datatype	Prec	Scale	Not ...	Key Type
1	DEALERSHIP_ID	number(p,s)	2	0		NOT A KEY
2	REGION	ar	7	0		NOT A KEY
3	CUSTOMER_NO	number(p,s)	5	0		NOT A KEY
4	FIRSTNAME	varchar	13	0		NOT A KEY
5	LASTNAME	varchar	15	0		NOT A KEY
6	ADDRESS	varchar	30	0		NOT A KEY
7	CITY	varchar	7	0		NOT A KEY

8) Scroll down and locate the field DATE_FLD.

   a) Change its datatype to "**Date**."

9) Click **OK**

10) Save your work.

Lab 2-3: Creating Mappings

Scenario:

- You need to create Mappings to connect Sources directly to Targets (*passthrough* Mappings) so that data can be loaded into the staging tables

Goals:

- Create shortcuts to objects in the shortcut (SC) folder
- Create a pass-through Mapping that brings data from a single Source to a single Target

Duration:

30 minutes

Instructions

Step 1. Create Shortcuts

Note: Best practices call for developers to build mappings from shortcuts to a common folder, rather than defining Sources and Targets in the developers' own folders. This has several advantages, of which the most significant is that it greatly eases migration of mappings between PowerCenter environments (e.g., from Development to Test to Production). Developers create sources and targets, and the Administrator copies them to the Shortcut folder, where they can be used by all developers, and in migration.

In this lab, you will use shortcuts based on the Sources and Targets you created in labs 1 and 2. The administrator has already copied these Sources and Targets. You will learn how to create shortcuts to objects in the shortcut folder.

Note: Best practices also call for data to be loaded directly from Sources into *staging* tables as part of the ETL process. From these tables, data can be accessed for transformation and loading without putting a further burden on the Source systems.

1) Make a shortcut to the Customers flatfile Source.

   a) In PowerCenter Designer, open the Source Analyzer workspace.

   b) If any Source definitions are showing, right-click the workspace and select **Clear All**

   c) Locate the folder SC_DATA_STRUCTURES.

Note: The SC_ prefix is a Velocity standard for shortcut folders.

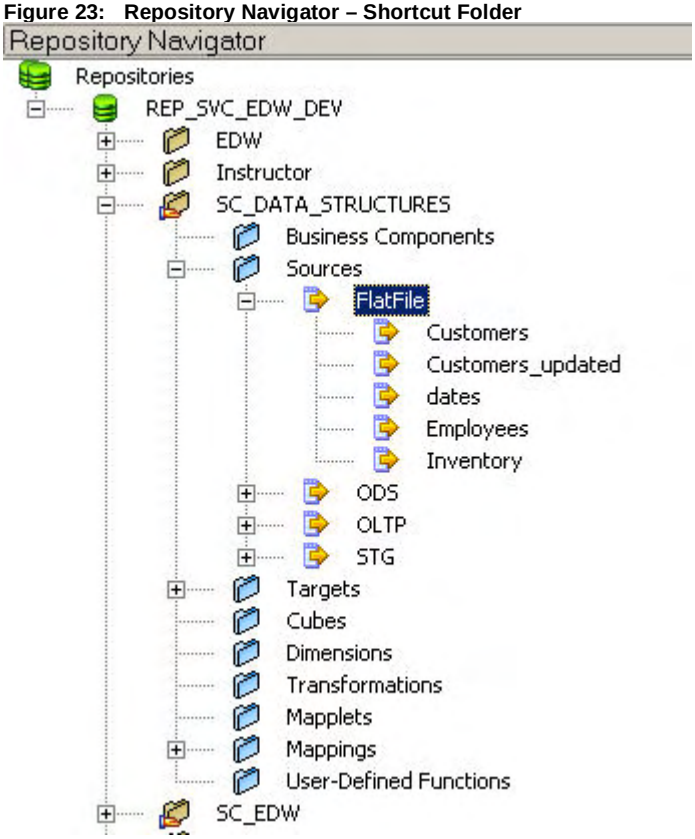
   d) Click once on the plus sign (+) to expand the folder.

Note: Do not double-click the name of the folder. This will connect you to the folder, and you need to remain connected to your own ~Developer***xx*** folder to create the shortcuts.

   e) Click once on the plus sign to the left of the subfolder named Sources.

f) Click once more on the plus sign to the left of the FlatFile subfolder. The Repository Navigator should now look like this:

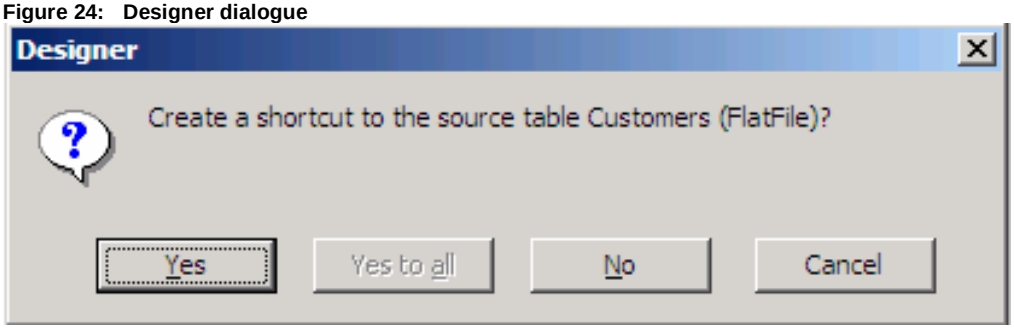
Figure 23: Repository Navigator – Shortcut Folder



g) Click the Customers flatfile Source definition and drag it into the Source Analyzer workspace.

h) In the Designer dialogue, click **Yes**.

Figure 24: Designer dialogue



Note: If the dialogue asks you whether to copy the source table, say No and try again. You want to make a shortcut, not a copy.

i) Double-click the Shortcut_to_Customers Source definition to edit it.

j) Click **Rename**.

k) Change the name of the source definition to **SC_Customers**.

l) Click **OK** to close the Edit Tables dialogue.

Note: The SC_ prefix is Velocity best practice for all shortcuts to objects in other folders.

m) In the Repository Navigator window, expand your ~Developer***xx*** folder, then the Sources sub-folder, then the FlatFile sub-subfolder.

n) Confirm that the shortcut appears there.

2) Following the same procedure, create a shortcut to the OLTP Source DEALERSHIP and rename it **SC_DEALERSHIP**.

3) Open the Target Designer.

4) Following the same procedure, create a shortcut to the Target STG_CUSTOMERS and rename it **SC_STG_CUSTOMERS**.

5) Following the same procedure, create a shortcut to the Target STG_DEALERSHIP and rename it **SC_STG_DEALERSHIP**.

6) Save your work.

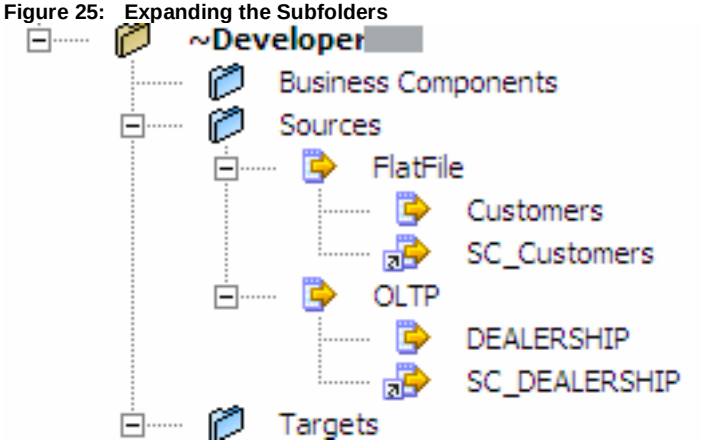
Step 2. Create the First Mapping

1) Click the Mapping Designer button () to open the Mapping Designer tool.

2) In this step, you will place all required components into the Mapping Designer workspace.

a) In your ~Developer***xx*** folder, expand the Sources subfolder, then the OLTP sub-subfolder.

Figure 25: Expanding the Subfolders



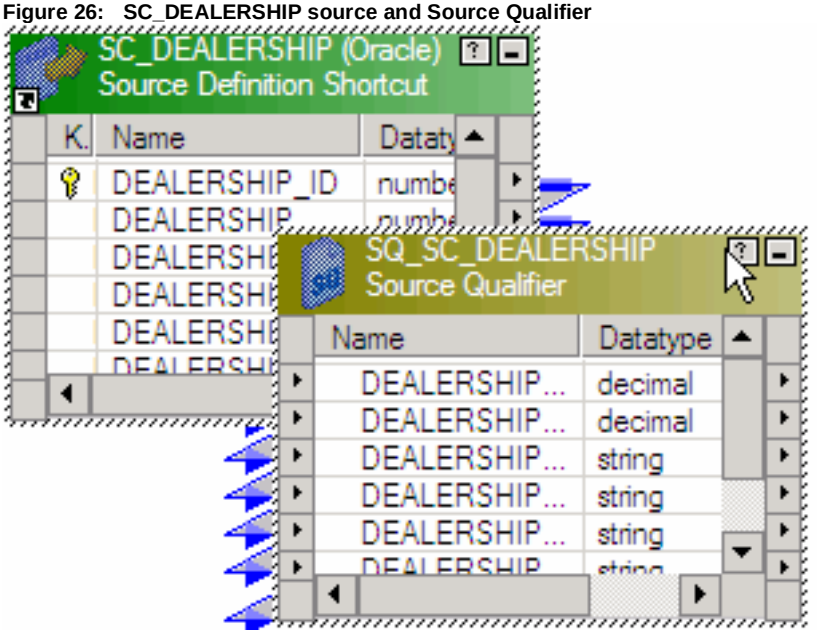
b) Drag the SC_DEALERSHIP shortcut onto the Mapping Designer workspace.

(i) You will be prompted to name the new Mapping. Give it the name **m2_STG_DEALERSHIP_*xx***. (Do not type "xx" – use your student id number!)

Note: Velocity best practice is for all Mappings to begin with the identifying prefix "m_"

(ii) Note that both a Source and a Source Qualifier transformation appear in the Mapping Designer. *If they did not, contact your instructor for help.*

Figure 26: SC_DEALERSHIP source and Source Qualifier



c) In your ~Developer*xx* folder, expand the Targets subfolder.

d) Drag the SC_STG_DEALERSHIP shortcut from the Targets folder onto the Mapping Designer workspace.

3) In this step, you will link the Source Qualifier to the Target.

Hint: This procedure may be easier if you rearrange the column widths in the Source Qualifier and Target so that you can see the full name of the port

a) In SQ_SC_DEALERSHIP, click **DEALERSHIP_ID** and keep your left mouse button depressed.

b) Drag the mouse pointer to SC_STG_DEALERSHIP, still keeping the button depressed.

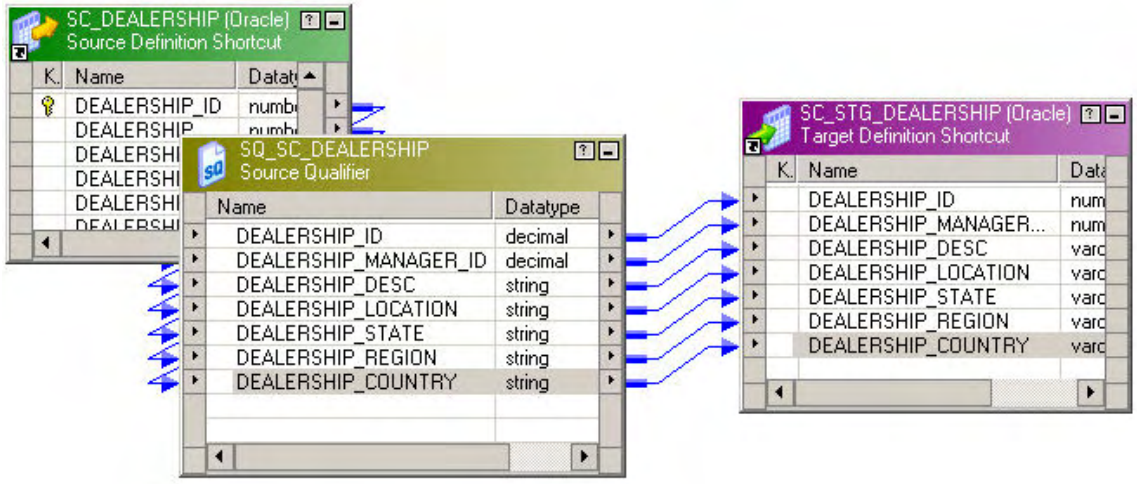
c) Hover over **DEALERSHIP_ID** on the target and release the mouse button.

d) A blue arrow, representing a link between the ports of the Source Qualifier and the Target definition, appears.

e) Repeat this process to link all ports of the Source Qualifier to the similarly-named ports in the Target.

f) The Mapping should look like this:

Figure 27: Completed mapping m2_STG_DEALERSHIP_*xx*



4) Save your work. In the Output Window, verify that the Mapping is valid. (If it is not, and you cannot spot the error, ask your instructor for help.)

Step 3. Create the Second Mapping

1) From the menu, select **Mappings → Create**.

   a) Name the new Mapping **m2_STG_CUSTOMERS_*xx***

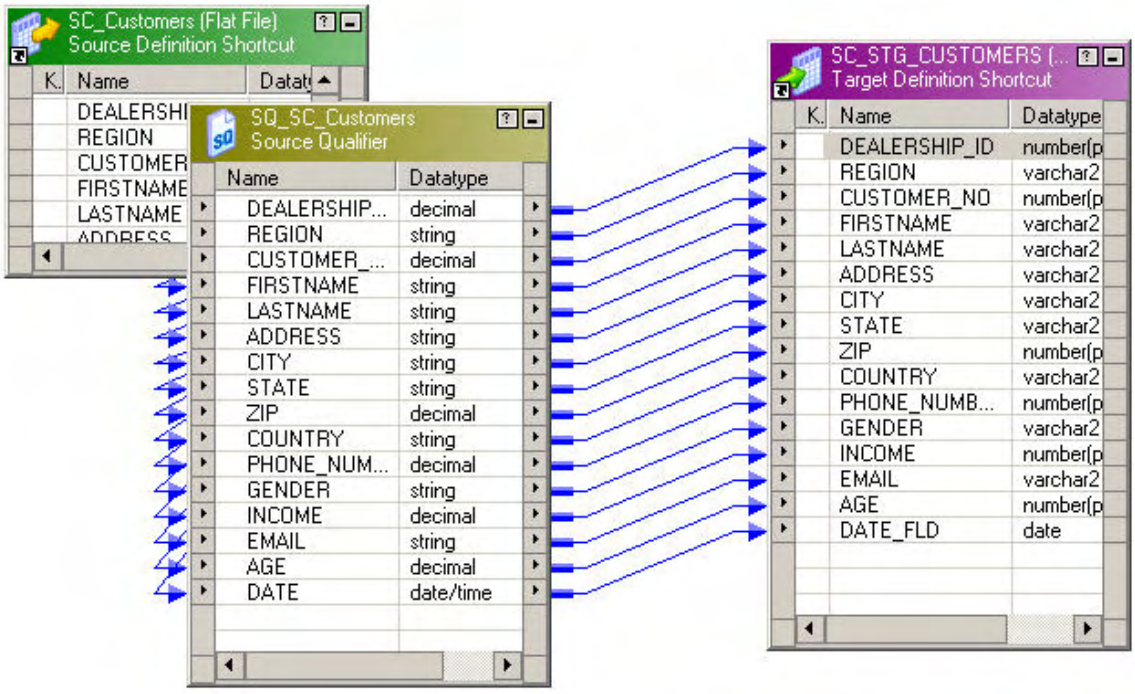
2) Drag the Source definition SC_Customers (the flat file definition, not the OLTP definition) and the Target definition SC_STG_CUSTOMERS into the Mapping Designer workspace.

   a) A Source Qualifier, SQ_SC_Customers, also appears.

   b) Using the process you have already learned, link the ports from the Source Qualifier to the same-named ports in the Target definition.

      (i) The DATE port in the Source Qualifier does not have a same-named port in the Target definition. Link it to the DATE_FLD port in the Target definition.

Figure 28: Completed mapping m2_STG_CUSTOMERS_*xx*



3) Save your work and verify that the mapping is valid.

Step 4. Create More Shortcuts

In the remaining labs of this class, you will use shortcuts to the objects in SC_DATA_STRUCTURES. It will be convenient to create those shortcuts now so they will be available in later labs.

1) In PowerCenter Designer, open the Source Analyzer.
2) Drag every Source definition in the SC_DATA_STRUCTURES folder onto the Source Analyzer workspace, except for Customers and DEALERSHIP (because you already have shortcuts to those).
3) Open the Target Designer.
4) Drag every Target definition in the SC_DATA_STRUCTURES folder onto the Target Designer workspace, except for STG_CUSTOMERS and STG_DEALERSHIP (because you already have shortcuts to those).
    a) **Be careful to create shortcuts, not copies!**

Note: The best practice is to change the names of each of these shortcuts to read "SC_" rather than "Shortcut_to_". However, it is time-consuming and dull.

The labs instructions from here on out will refer to these shortcuts using the "SC_" prefix, to reflect the best practice; if you do not change the names, simply substitute "Shortcut_to_".

Lab 3-1: Create and Run Workflows

Scenario:

- You need to load the Customer and Dealership data into the Staging tables.

Goals:

- Create and run Workflows that execute the Mappings you created in Lab 2-3

Duration:

45 minutes

Instructions

Step 1. Open Workflow Manager

1) In PowerCenter Designer, locate the "Tools" toolbar.

Figure 29: Tools Toolbar

2) Click the "W" icon to open the Workflow Manager application.

Step 2. Create a Workflow

1) If necessary, connect to your ~Developer*xx* folder.
2) From the Workflow Manager menu, select **Tools → Workflow Designer** (**NOT Workflow Manager**).
3) From the Workflow Manager menu, select **Workflows → Create**.
4) In the Create Workflow dialogue
   a) Name the Workflow **wkf_STAGE_DEALERSHIP_*xx***

> **Note:** Velocity best practice is to prefix the name of a Workflow with "wkf_" and give it a name describing what it does. In this case, we are loading dealership data into a staging table.

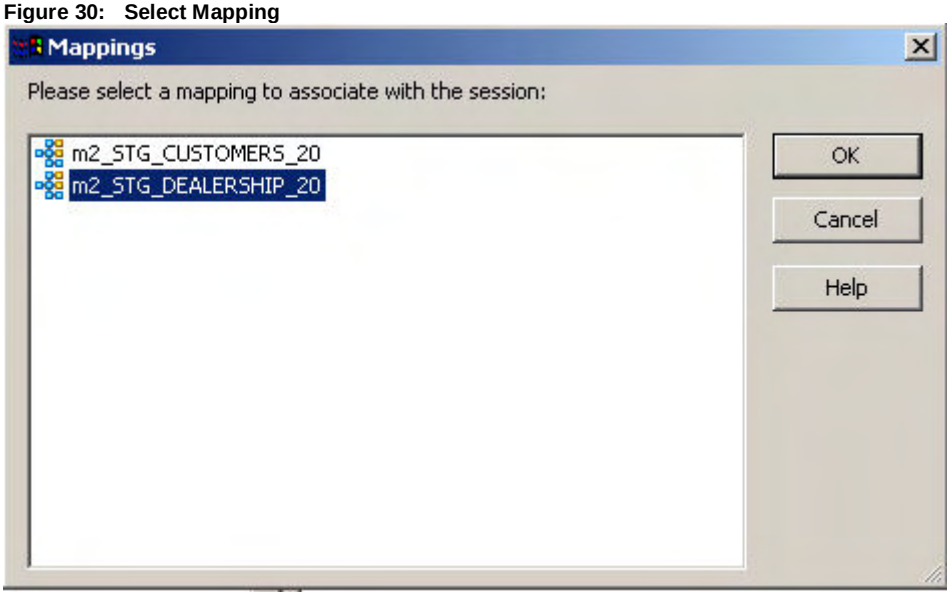
   b) Click **OK**.
   c) Note that the Workflow is created with a Start task already present.

5) In the toolbar, click the Session button ( ).

a) Click in the Workflow Designer workspace somewhere to the right of the Start task.

b) In the Mappings window, select m2_STG_DEALERSHIPS_*xx* and click **OK**.

Figure 30: Select Mapping



c) The Session task is added to the Workflow:

Figure 31: Session Added

Note: The Velocity standard name for a Session task is **s_** followed by the name of the Mapping. The Workflow Designer automatically assigns this name to a Session task when you add it to the Workflow.

6) From the Workflow Manager menu, select **Tasks→Link Task**

   a) Place the mouse curser over the Start task.

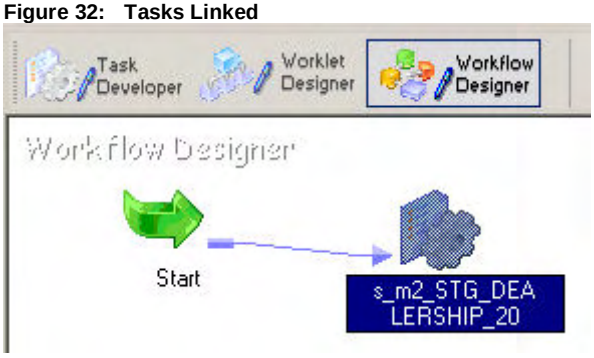
   b) Click and hold the left mouse button.

   c) Drag the cursor to the Session task.

   d) Release the mouse button.

   e) The tasks will be linked:

Figure 32: Tasks Linked



7) Double-click the Session task to edit its properties.

Note: The Session task properties determine what files or database tables the Mapping reads from and writes to. The Source and Target definitions in the Mapping define the fields to be read or written, but do not directly determine where to read from or write to.

a) Select the Mapping tab of the Edit Tasks window.

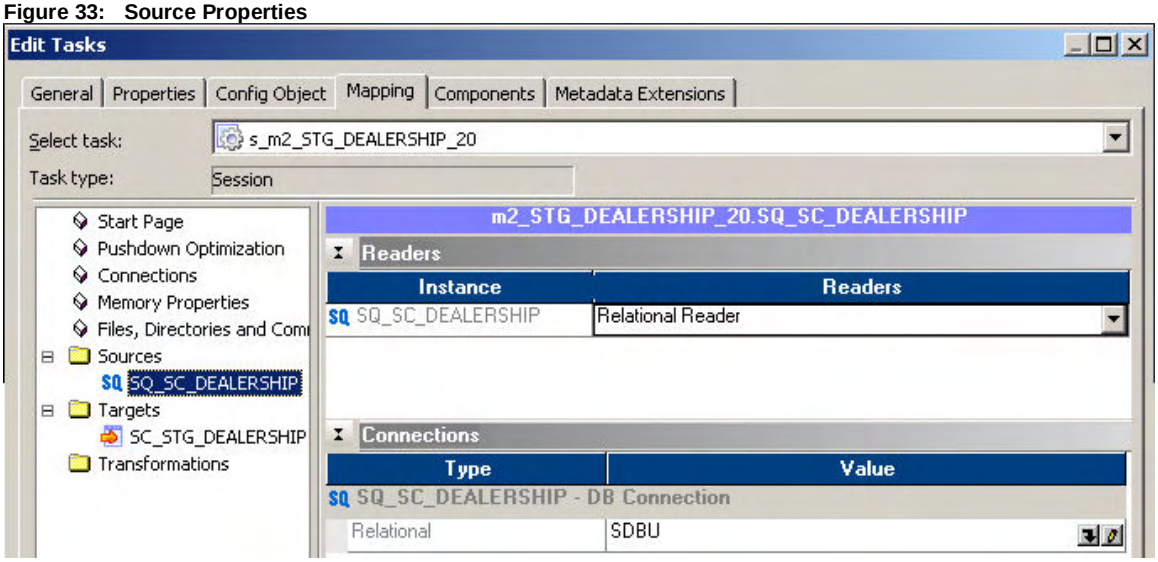
b) In the navigator on the left, select the Source **SQ_SC_DEALERSHIP**.

c) On the right:

(i) Ensure that the "Reader" selected is **Relational Reader**.

(ii) In the "Connections" section, click the dropdown arrow and select the connection **SDBU**.

Figure 33: Source Properties



d) In the navigator on the left, select the Target **SC_STG_DEALERSHIP**.

e) On the right:

(i) Ensure that the "Writer" selected is **Relational Writer**.

(ii) Ensure that the Relational connection selected is **STG*xx***.

(iii) In the "Properties" section, set the "Target load type" to **Normal**.

(iv) Scroll down and check "Truncate target table option".

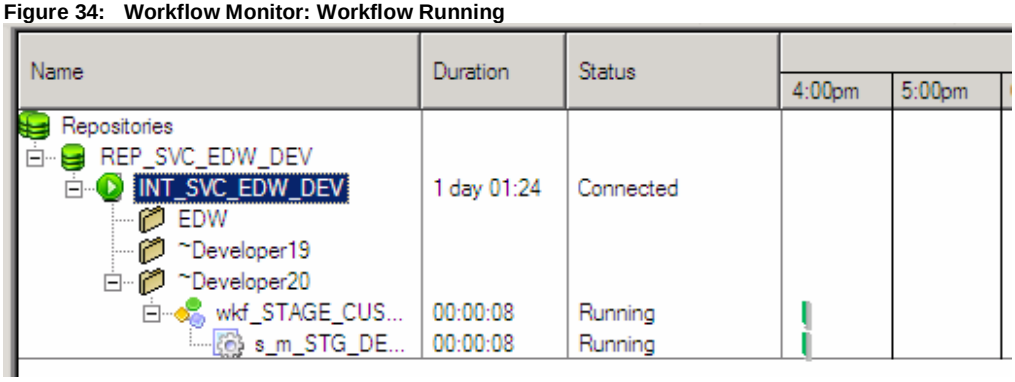
(v) Click **OK**.

8) Save your work.

Step 3. Start the Workflow.

1) In the "Tools" toolbar, click the "M" icon to start the Workflow Monitor application.
2) If the status for "INT_SVC_EDW_DEV" reads "Disconnected", right-click it and select **Connect**.
3) Return to the Workflow Manager.
4) Right-click in the Workflow Designer workspace and select **Start Workflow**.
5) Return to the Workflow Monitor. The status of your Workflow and Session are now both "Running."

Figure 34: Workflow Monitor: Workflow Running

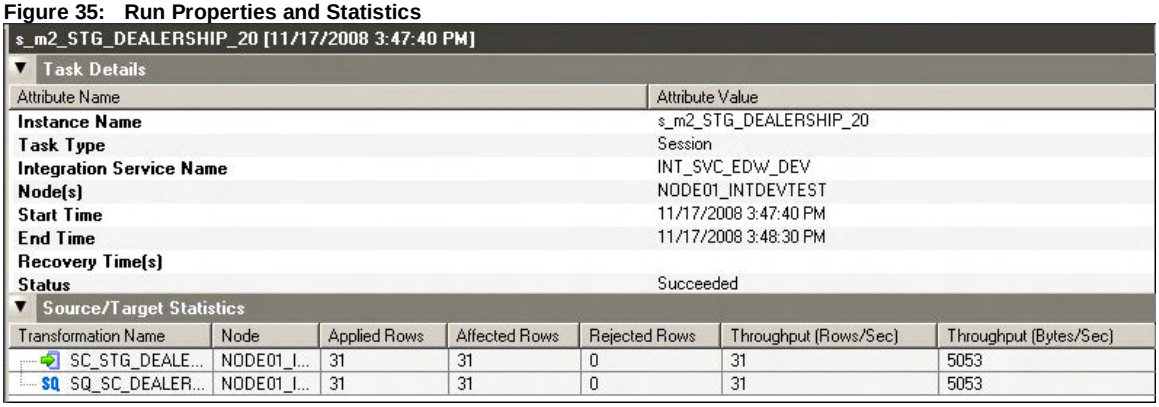


6) When the status changes to "Succeeded" your Workflow has run.

Step 4. View the Results

1) Right-click the Session object and select "Get Run Properties."
2) Click the arrow beside "Source/Target Statistics" to expand it.

Figure 35: Run Properties and Statistics



3) Return to the PowerCenter Designer application.

4) In the m_STG_DEALERSHIPS_*xx* Mapping, right-click the Target object and select **Preview Data**.

5) In the Preview Data dialogue:

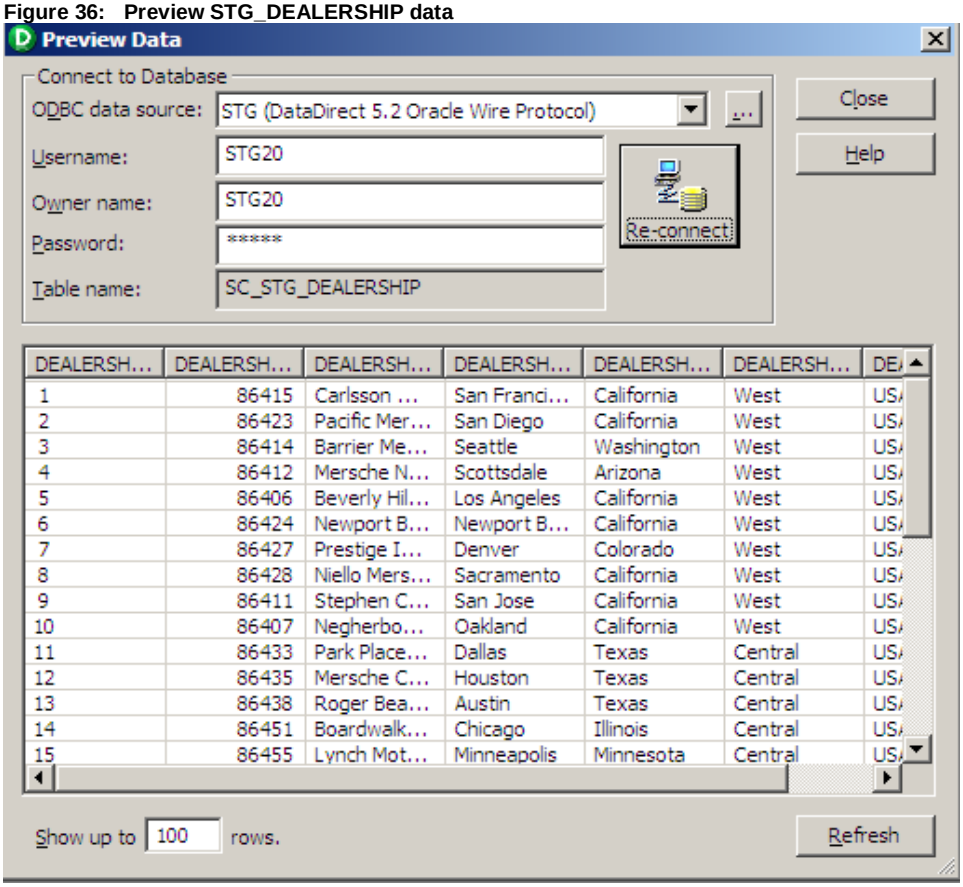
   a) For ODBC Data Source, select **STG (DataDirect 5.2 Oracle Wire Protocol)**.

   b) For Username, Table Owner, and Password, enter **STG*xx***

   c) Click **Connect**.

   d) The lower part of the Preview Data dialogue populates with the data you loaded into the STG_DEALERSHIP table.

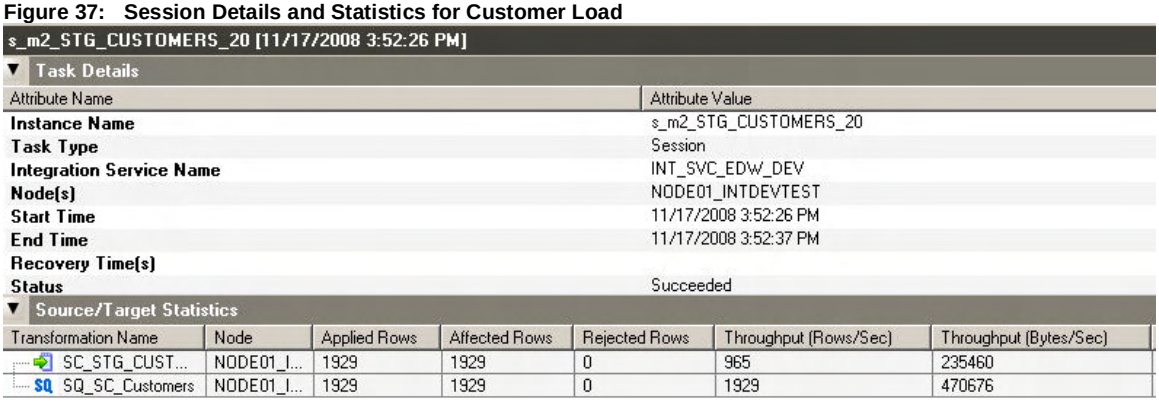
Figure 36: Preview STG_DEALERSHIP data



Step 5. Create and Run a Second Workflow

1) Following the same procedure, create a second Workflow.
   a) Name the Workflow **wkf_STAGE_CUSTOMERS_*xx***.
   b) Create a Session task using the Mapping **m2_STG_CUSTOMERS_*xx***.
   c) Link the Start task to the Session task.
2) Double-click the Session task to edit it.
   a) Select the Mapping tab.
   b) Select the source **SQ_SC_Customers**
      (i) Make sure that the reader type is "File Reader"
      (ii) Make sure that the Source Filetype is "Direct"
      (iii) Make sure that the Source file directory is "$PMSourceFileDir\"
      (iv) Make sure that the Source filename is "customer_central.dat"
   c) Select the target **SC_STG_CUSTOMERS**
      (i) Make sure that the Writer type is "Relational Writer"
      (ii) Make sure that the Connection is "STG***xx***"
      (iii) Make sure that the Target load type is "Normal"
      (iv) Make sure that the "Truncate target table option" is checked.
   d) Click **OK**.
3) Save your work and make sure the Workflow is valid.
4) Run the Workflow and review the results.

Figure 37: Session Details and Statistics for Customer Load



s_m2_STG_CUSTOMERS_20 [11/17/2008 3:52:26 PM]

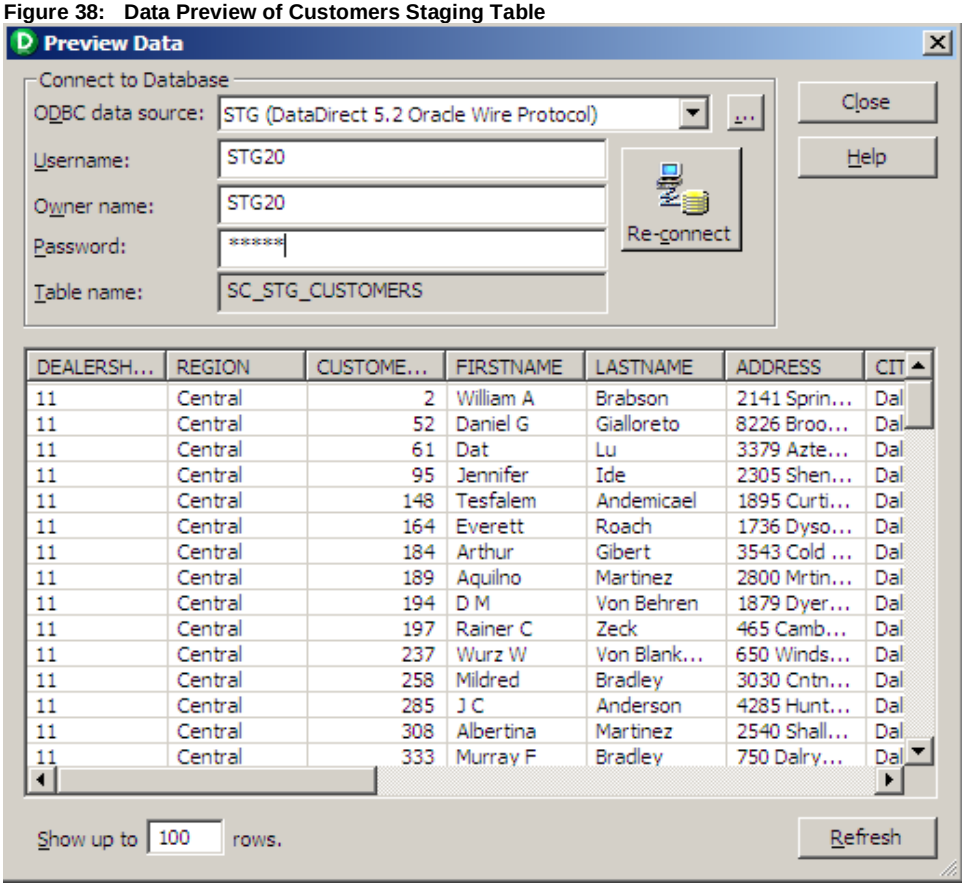
Task Details

Attribute Name	Attribute Value
Instance Name	s_m2_STG_CUSTOMERS_20
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	NODE01_INTDEVTEST
Start Time	11/17/2008 3:52:26 PM
End Time	11/17/2008 3:52:37 PM
Recovery Time(s)	
Status	Succeeded

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_STG_CUST...	NODE01_I...	1929	1929	0	965	235460
SQ_SC_Customers	NODE01_I...	1929	1929	0	1929	470676

Figure 38: Data Preview of Customers Staging Table



Extra Credit Lab: File Lists

You may have noted that the name of the Customers file you loaded was "customer_central.dat". There are two other files, "customer_east.dat" and "customer_west.dat"

You could load each of these files individually, but there is an easier way: use a file list. PowerCenter can take as its input a text file listing other files, and will read each of the files in that list.

Note that PowerCenter will read all the files in the file list using the same Source definition. This means that the files must have ***exactly*** the same format, or PowerCenter will read bad data.

To make this work, you must make the following changes to the Session object:

- The Source Filetype has to be set to "indirect," indicating that the Source file is a file list rather than a data file.
- The Source Filename must be changed to the name of the file containing the file list, which is "customer_list.dat"

Lab 4-1: Using Filters and Expressions

Scenario:

- The customer contacts have been loaded to the Staging table and are ready to be loaded to ODS
- There are bad customer IDs that should not be loaded to ODS
- The customer data needs to be reformatted to match the table specification of the ODS

Goals:

- Move data from the Customer staging table to the ODS database
  - Use an Expression transformation to reformat data
  - Use a Filter transformation to pass only valid records

Duration:

60 minutes

Instructions

Step 1. Create the mapping.

1) Create a new mapping called **m4_ ODS_CUSTOMERS_*xx***.
2) Add the Source definition SC_STG_CUSTOMERS to the mapping.
3) Add the Target definition SC_ODS_CUSTOMERS to the mapping.
4) The Mapping should look like this:

Figure 39: Mapping Created

Step 2. Add a Filter Transformation

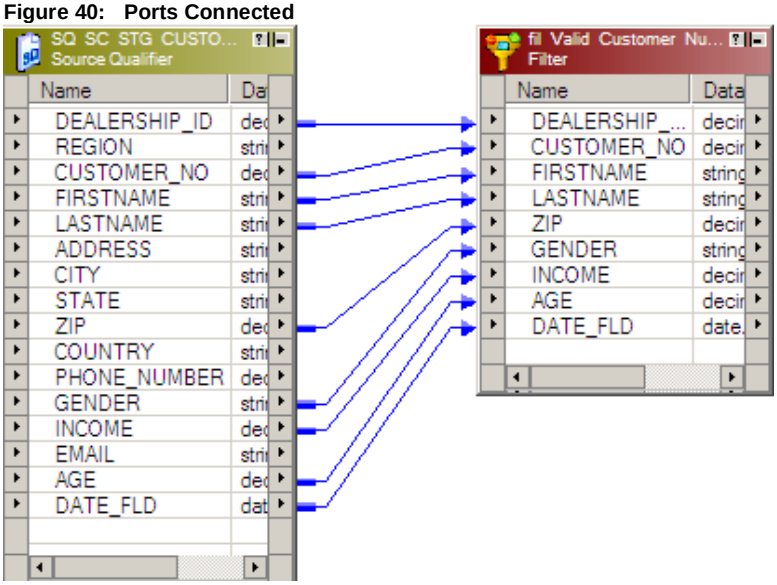
In this step you will add a Filter transformation to the mapping to pass only records with valid Customer IDs.

1) Locate the Filter transformation ( ) on the toolbar. Click it and then click on the Mapping Designer workspace.

2) Drag the following ports from the Source Qualifier to the Filter transformation:

DEALERSHIP_ID
CUSTOMER_NO
FIRSTNAME
LASTNAME
ZIP
GENDER
INCOME
AGE
DATE_FLD

Figure 40: Ports Connected



3) Double-click the header of the Filter transformation to edit it.

4) Rename the filter object **fil_Valid_Customer_Number**.

5) Select the Properties tab.

   a) Click the bent arrow ( ) in the Value column for the attribute Filter Condition.

   b) In the Expression Editor:

      (i) Delete the current expression ("TRUE") from the Expression Editor.

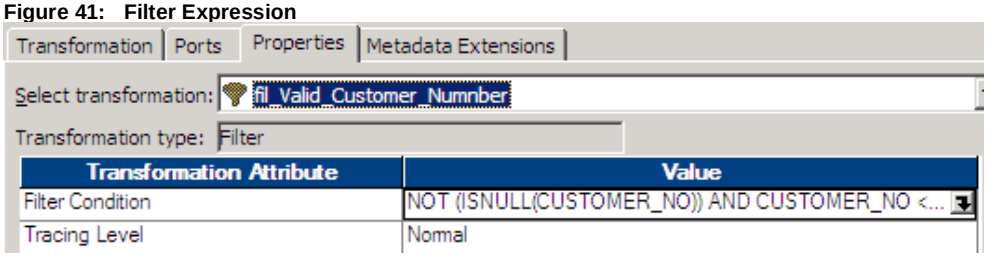
      (ii) Enter the expression **NOT (ISNULL(CUSTOMER_NO)) AND CUSTOMER_NO != 99999**

      (iii) Click **Validate**

      (iv) If the expression is invalid, fix it.

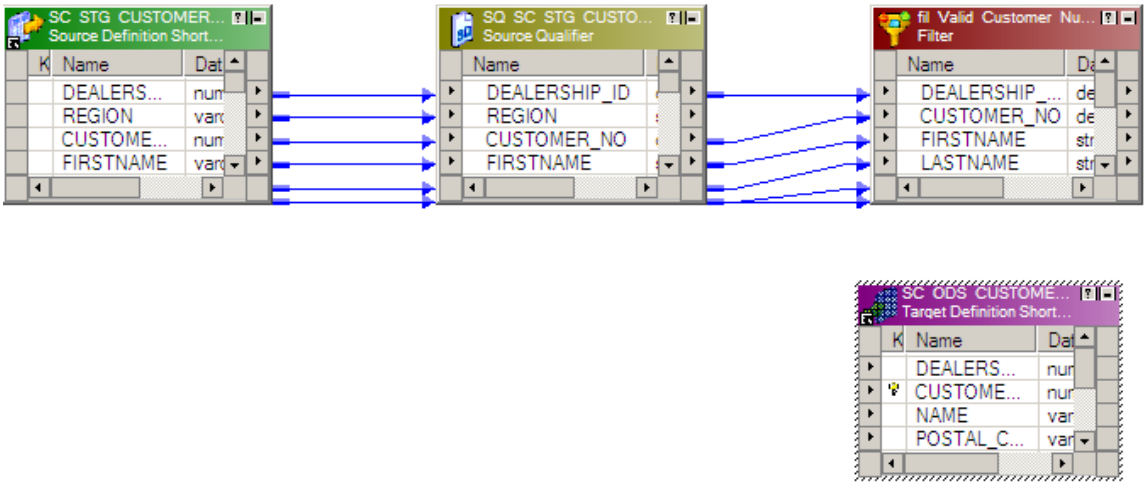
      (v) Click **OK**.

Figure 41: Filter Expression



   c) Click **OK**.

Figure 42: Mapping with Filter Transformation Added



Step 3. Add an Expression Transformation

In this step you will add an Expression transformation that will format Customer data correctly for the ODS database.

1) Locate the Expression transformation ( ) on the toolbar. Click it and then click on the Mapping Designer workspace.

2) Drag the following ports from the Filter transformation to the Expression transformation:

FIRSTNAME
LASTNAME
GENDER
INCOME
AGE

3) Edit the Expression transformation

a) Rename the Expression transformation **exp_Format_Customers**.

b) Select the Ports tab.

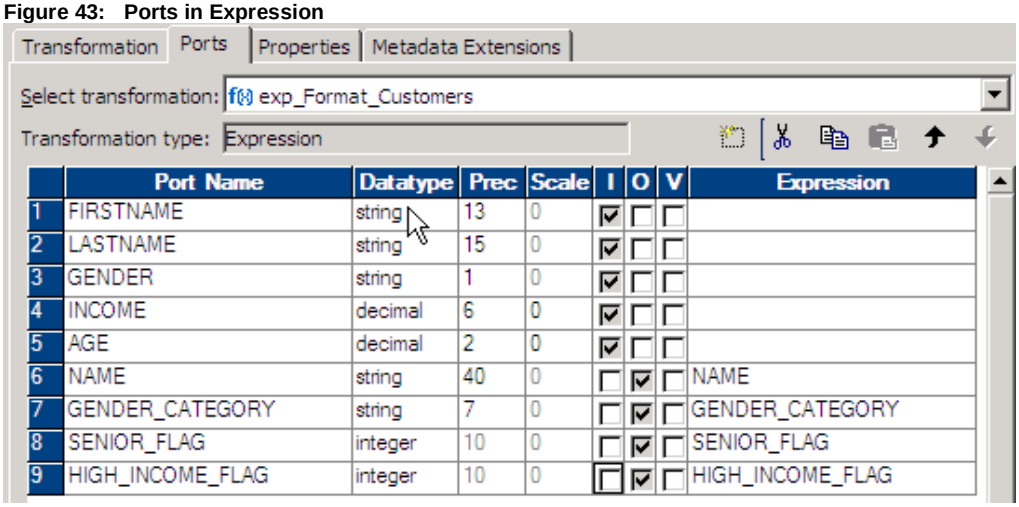
c) Set all the ports you dragged from the Filter to be input-only by unchecking the "O" column.

d) Use the New button ( ) to create the following new ports and set them to output only:

Port Name	Datatype	Length/Precision
NAME	string	40
GENDER_CATEGORY	string	7
SENIOR_FLAG	integer	10
HIGH_INCOME_FLAG	integer	10

e) The ports should now look like this:

Figure 43: Ports in Expression



	Port Name	Datatype	Prec	Scale	I	O	V	Expression
1	FIRSTNAME	string	13	0	☑	☐	☐	
2	LASTNAME	string	15	0	☑	☐	☐	
3	GENDER	string	1	0	☑	☐	☐	
4	INCOME	decimal	6	0	☑	☐	☐	
5	AGE	decimal	2	0	☑	☐	☐	
6	NAME	string	40	0	☐	☑	☐	NAME
7	GENDER_CATEGORY	string	7	0	☐	☑	☐	GENDER_CATEGORY
8	SENIOR_FLAG	integer	10	0	☐	☑	☐	SENIOR_FLAG
9	HIGH_INCOME_FLAG	integer	10	0	☐	☑	☐	HIGH_INCOME_FLAG

f) Use the Expression Editor to create an expression for the NAME port to concatenate the FIRSTNAME and LASTNAME fields, with a space in between:
FIRSTNAME || ' ' || LASTNAME

Note: More advanced data integration developers may recognize that the above expression leaves something to be desired when dealing with less-than-ideal data, as would be typical in these fields. Informatica has extensive data quality capabilities to recognize, cleanse, and supplement name data. These capabilities are in the Data Quality product, which is outside the scope of this class.

g) Create an expression for the GENDER_CATEGORY port that expands single character designations from the GENDER field into full-word descriptions:
DECODE(GENDER, 'M', 'MALE', 'F', 'FEMALE', 'UNK')

The DECODE function uses a mapping to replace the values in a field with other values. In this case, if the field has a value of "M", then it is changed to "MALE." If the field has a value of "F", it is changed to "FEMALE." Any other value will be replaced with "UNK" (for "UNKNOWN"). DECODE is useful when there are a relatively small number of enumerated values in a field. If there are a larger number of values to be remapped, a Lookup transformation would be used. (We will cover Lookup transformations later in this course.)

h) Create an expression for the SENIOR_FLAG port that sets the port value to 1 (Boolean TRUE) if the AGE is greater than 55:
IIF(AGE > 55, 1)

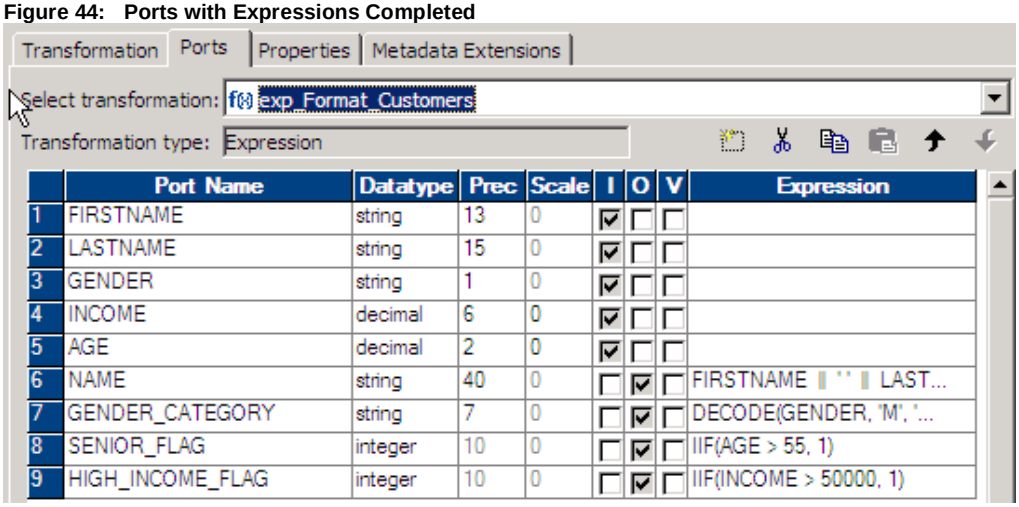
IIF – "Immediate If" – is a powerful function. When the expression (AGE > 55) evaluates to TRUE, the first argument is assigned to the port. When the expression does not evaluate to TRUE, the second argument is assigned to the port. In this case, no second value is assigned, so the port is set to zero (0) when the expression evaluates to FALSE.

IIF expressions can be nested to handle multibranch logic.

i) Create an expression for the HIGH_INCOME_FLAG port that sets the port to 1 (Boolean TRUE) if the INCOME is greater than 50000:
IIF(INCOME > 50000, 1)

j) The ports should now look like this:

Figure 44: Ports with Expressions Completed



	Port Name	Datatype	Prec	Scale	I	O	V	Expression
1	FIRSTNAME	string	13	0	☑	☐	☐	
2	LASTNAME	string	15	0	☑	☐	☐	
3	GENDER	string	1	0	☑	☐	☐	
4	INCOME	decimal	6	0	☑	☐	☐	
5	AGE	decimal	2	0	☑	☐	☐	
6	NAME	string	40	0	☐	☑	☐	FIRSTNAME || ' ' || LAST...
7	GENDER_CATEGORY	string	7	0	☐	☑	☐	DECODE(GENDER, 'M', '...
8	SENIOR_FLAG	integer	10	0	☐	☑	☐	IIF(AGE > 55, 1)
9	HIGH_INCOME_FLAG	integer	10	0	☐	☑	☐	IIF(INCOME > 50000, 1)

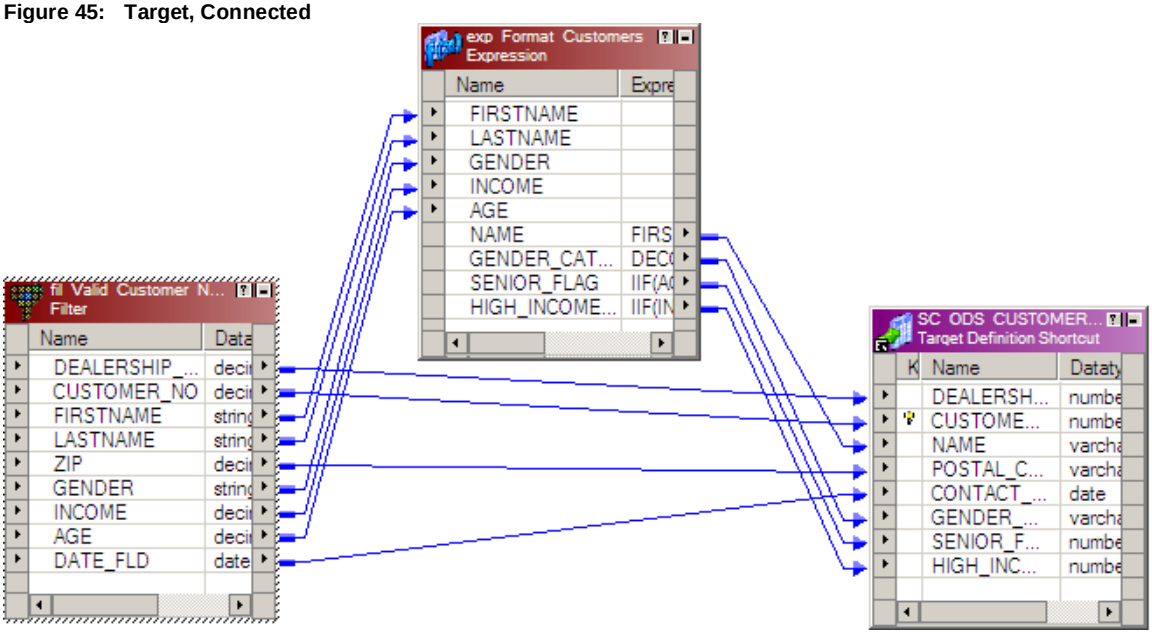
4) Click **OK**.

Step 4. Connect the Ports to Complete the Mapping

1) Connect the following ports:

From Transformation Name	From Port Name	To Transformation Name	To Port Name
fil_Valid_Customer_Number	DEALERSHIP_ID	SC_ODS_CUSTOMER	DEALERSHIP_ID
fil_Valid_Customer_Number	CUSTOMER_NO	SC_ODS_CUSTOMER	CUSTOMER_NO
fil_Valid_Customer_Number	ZIP	SC_ODS_CUSTOMER	POSTAL_CODE
fil_Valid_Customer_Number	DATE_FLD	SC_ODS_CUSTOMER	CONTACT_DATE
exp_Format_Customers	NAME	SC_ODS_CUSTOMER	NAME
exp_Format_Customers	GENDER_CATEGORY	SC_ODS_CUSTOMER	GENDER_CATEGORY
exp_Format_Customers	SENIOR_FLAG	SC_ODS_CUSTOMER	SENIOR_FLAG
exp_Format_Customers	HIGH_INCOME_FLAG	SC_ODS_CUSTOMER	HIGH_INCOME_FLAG

Figure 45: Target, Connected

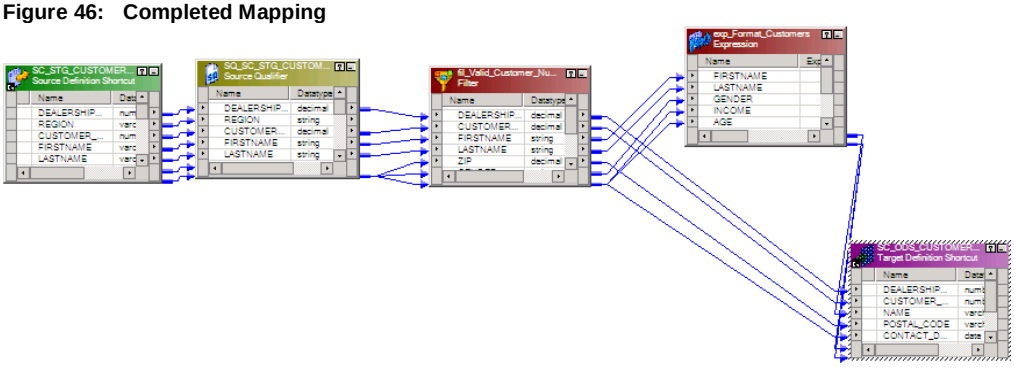


2) Save your work.

   a) Verify that the Mapping is valid and fix any problems that keep it from validating.

3) The Mapping should look like this:

Figure 46: Completed Mapping



Step 5. Create and Run the Workflow

1) Open PowerCenter Workflow Manager.
2) Create a new workflow named wkf_Load_ODS_CUSTOMERS_*xx*.
3) Add a Session task using the mapping m4_ODS_CUSTOMERS_*xx*.
4) Edit the Session task and set the connection values:
   a) Click the Mapping tab
   b) Source connection should be STG*xx*
   c) Target connection should be ODS*xx*
   d) Make sure to set the Target load type to Normal
   e) Make sure to set the Truncate target table option
5) Link the Session to task the Start task
6) Save your work.
7) Start the workflow and monitor the results.

Figure 47: Results of Running the Workflow

s_m4_ODS_CUSTOMERS_20 [11/17/2008 4:20:26 PM]

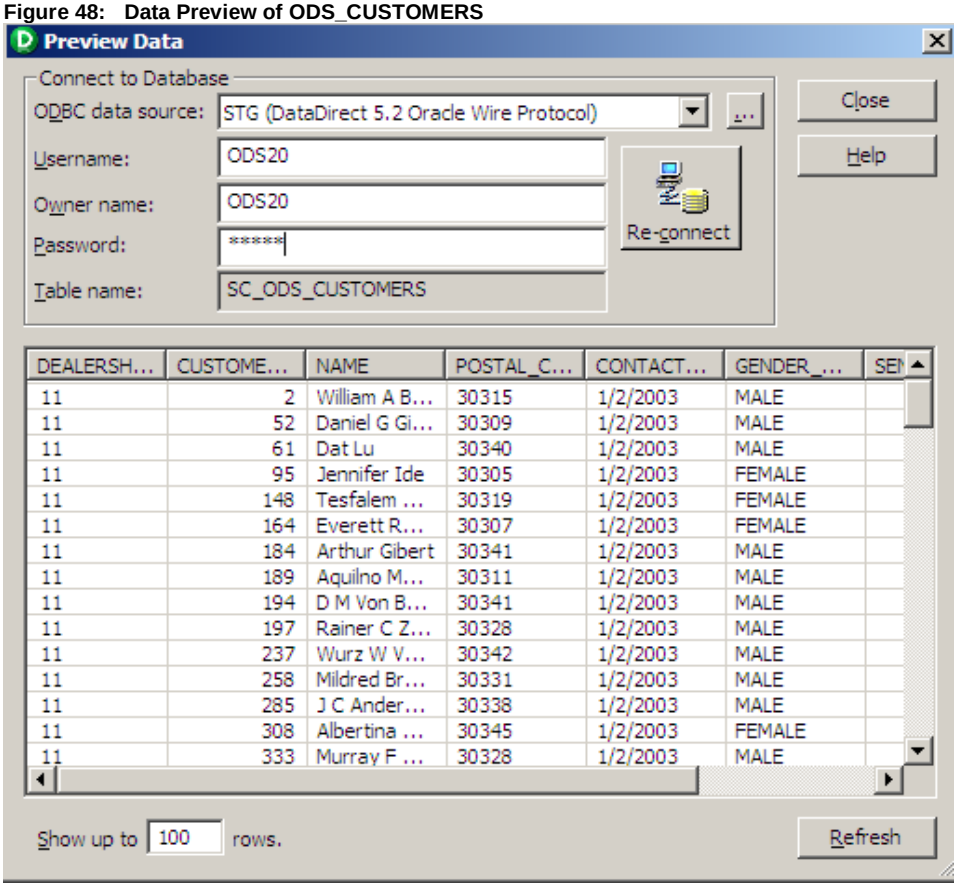
Task Details

Attribute Name	Attribute Value
Instance Name	s_m4_ODS_CUSTOMERS_20
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	NODE01_INTDEVTEST
Start Time	11/17/2008 4:20:26 PM
End Time	11/17/2008 4:20:37 PM
Recovery Time(s)	
Status	Succeeded

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_CUST...	NODE01_I...	1924	1924	0	1924	198172
SQ_SC_STG_CU...	NODE01_I...	1929	1929	0	1929	169752

Figure 48: Data Preview of ODS_CUSTOMERS



Note: The number of rows you see may differ from what is shown in the figure, depending on whether you performed the extra credit exercise at the end of Lab 4.

a) Why does the number of rows in the source not match those in the target?

Answers

5.7.a. Why does the number of rows in the source not match those in the target?
Some rows were removed by the Filter transformation, so those rows did not reach the target.

Lab 4-2: Features and Techniques

Goals:

- In this lab you will learn and practice some features and techniques that will increase your efficiency as a PowerCenter Developer.

Duration:

30 minutes

Instructions

> **WARNING:** In this lab, ***do not*** save your work. While it is normally best practice to save your work frequently while working in PowerCenter, in this case you will be making changes to a Mapping that is already the way you want it. So don't save your work!

Step 1. Arrange All and Arrange All Iconic

> In a complex Mapping, it can be hard to see how the parts relate. How can you make this better?

1) Begin with the Mapping from Lab 4-1 (m4_ODS_Customers_***xx***) open in the PowerCenter Developer application.
2) Right-click anywhere in the workspace and select **Arrange All**.
3) Observe the results.

> Arrange All is a tool for arranging the transformations in a Mapping neatly.

4) Right-click again and select **Arrange All Iconic**.

> Arrange All Iconic enables you to quickly see the relationships between the objects in a Mapping.

Step 2. Autolink

1) "Arrange All" on the Mapping.
2) Drag the cursor across the links between the Source definition and the Source Qualifier to select them.
3) Hit the Delete key on your keyboard.
4) Right-click and select **Autolink by Name**.
5) Position the cursor over the Source, then click and drag to the Source Qualifier.
6) Click the SQ again to return to the normal (arrow) cursor.

> Autolinking provides a quick way to connect the output ports in one transformation to the input ports in another transformation.
>
> Autolink by Name searches for ports with identical names and connects them
>
> Autolink by Position connects the first output port to the first input port, the second output port to the second input port, etc.

7) Delete the links again and Autolink the two by Position.

Step 3. Select Link Path

Suppose another developer has created a large, complex Mapping that is not working quite right: some data is winding up in the wrong fields. And you have been asked to debug it. How can you figure out where the data is coming from? Answer: By tracing the link paths.

a) On the Target definition, right-click the POSTAL_CODE field and select **Select Link Path → Backward**.

(i) The link to the Postal Code field is now red.

b) Expand the Filter transformation so you can see the related field there.

(i) Note that the links leading both into and out of it are red.

c) You can, by expanding the appropriate transformations, trace the lineage of the Postal Code field all the way back to the ZIP field in the Source definition.

Selecting the link path enables you to easily trace the lineage of any field forward and backward through a Mapping.

Step 4. Propagating Port Properties

You have to change the datatype of a field in the Source. Do you really have to manually adjust every port along its link path? No.

1) Edit the Source Qualifier and select the Ports tab.

2) Change the name of the CUSTOMER_NO port to CUST_NO and its precision from 5 to 10.

3) Click **OK**.

4) Right-click CUST_NO in the Source Qualifier and select Propagate Attributes.

5) In the "Propagate Port Attributes" dialogue:

a) Under "Attributes to Propagate" select Name and Precision, with a direction of Both.

b) Click **Preview**.

c) Note the green and red arrows. What will be changed?

d) Click **Propagate**, then **Close**.

e) Was a change made in the Filter? What was it?

f) Was a change made in the Target definition? Why or why not?

Step 5. Moving Ports

Sometimes just rearranging the ports on a transformation will make the Mapping easier to read.

1) Edit the Filter transformation and select the Ports tab.
2) Click the AGE port and use the "up arrow" () button to move it to the top of the list of ports.
3) Single-click and hold the number next to the ZIP field. Note the square that appears in the cursor.
4) Drag ZIP right below AGE.
5) Click Cancel to discard the changes.

Step 6. Another Method of Creating Transformations

Plus, it bypasses the default names PowerCenter gives a transformation.

1) From the menu, select **Transformation→Create**
2) Select Aggregator from the dropdown box.
3) Name the Aggregator **agg_Demo_Create**.
4) Click **Create**.
   a) The new transformation appears in the workspace.
5) By the same method, create a Filter named **fil_Demo_Create**.
6) Click **Done**.

While we're at it, how do you remove an unwanted transformation?

7) The Filter you just created is already selected. Hold down the Shift key and click the Aggregator you created to select it, too.
8) Hit the Delete key on your keyboard.
   a) Note that the Designer dialogue tells you which transformations will be deleted.
   b) Click **Yes**.

Step 7. Reverting to Saved

Sometimes you make a mistake that you can't easily undo and need to go back to where you were before. If you haven't saved, you can do it.

1) In the Repository Navigator, right-click your folder and select **Disconnect**.
2) When asked whether to save the changes to your folder, click **No**.
3) Reopen your folder.
4) If necessary, reopen the Mapping.
5) Note that it is back to the way it was before Step 1.
6) Arrange All for the next step.

Step 8. Scaling

You may not be able to see the whole Mapping in your workspace. But you can.

1) Maximize PowerCenter Developer.
2) How many transformations can you see?
3) In the Standard toolbar at the top of the window, click the Zoom dropbox () and select 60.
4) Can you see more transformations?
5) Click the Scale to fit icon () beside the Zoom dropbox.
6) You can now see all your transformations at once.

Step 9. Switching Transformations While Editing

When editing several transformations, you don't have to close the Edit Transformations dialogue and reopen it repeatedly…

1) Double-click the Source Qualifier transformation to edit it.
2) Select the Ports tab.
3) In the Select transformation: dropbox, select the Filter transformation.
4) What happens?

Step 10. Copy Objects Within and Between Mappings

You may find that you want to duplicate a set of transformations within a Mapping or a Mapplet, preserving the dataflow between them. This technique may prove useful if you know that you will need to use the logic contained in the transformations in other Mappings or Mapplets.

1) Arrange All Iconic.
2) Use your left mouse button to draw a rectangle that encloses the Filter and Expression transformations. This will select these objects.
3) Press Ctrl+C on your keyboard, immediately followed by Ctrl+V.
4) Note that both transformations have been copied onto the mapping, including the dataflow between them. They have been renamed with a "1" on the end of their names.
5) Open another Mapping.
6) Press Ctrl+V again.
7) The transformations are added to the open Mapping.
8) Disconnect from your folder but ***do not*** save the changes (revert to the previously saved version).

Step 11. View Object Dependencies

By viewing object dependencies in the Designer, a user can learn which objects may be affected by making changes to Source or Target definitions, Mappings, Mapplets, or transformations. Direct and indirect dependencies are shown.

1) In the Repository Navigator, select the flat-file Source definition SC_Customers.
2) Right-click and select Dependencies.
3) Click **OK** to show all dependencies.
4) You will see the View Dependencies window, which will show every Mapping, Session, and Workflow that uses or depends upon the SC_Customers Source, as well as those that it uses or depends on.
5) It also shows the Customers flat file definition in SC_DATA_STRUCTURES that SC_Customers depends on.

Note: The Save to File button on the View Dependencies window saves the dependency information as a HTML file (.htm) for later viewing.

6) Experiment by viewing the dependencies of other objects.

Answers

4.5.e. Was there a change made in the Filter? What was it?
Yes, the name and precision of the Customer Number port changed to match the changes in the Source Qualifier.

4.5.f. Was there a change made in the Target definition? Why or why not?
No, the Source and Target definitions cannot be changed or edited in the Mapping Designer workspace. They can only in the Source Analyzer and Target Designer workspaces.

9.4. What happens?
You now see and can work with the ports of the Filter transformation.

Lab 5-1: Joining Data

Scenario:

- The ODS requires a Personnel table containing information about employees and the dealership each works at.
  - Personnel information is stored in two staging tables named STG_EMPLOYEES and STG_DEALERSHIP
- The ODS also requires a Stock Units table containing information about how many and what kind of units are in stock at the dealerships
  - The Stock Units information is contained in two sources as well, STG_PRODUCT and a flatfile named Inventory.

Goals:

- Create a Mapping with two separate data flows.
- Join STG_EMPLOYEES and STG_DEALERSHIP using a single Source Qualifier. Set the join condition for the tables.
- Make a previously-created transformation reusable, and use it in this Mapping.
- Join Inventory and STG_PRODUCT using a PowerCenter Joiner transformation.
- Create and execute a Workflow to populate both ODS_Personnel and ODS_StockUnits.

Duration:

90 minutes

Instructions

Step 1. Create a Homogeneous Join

1) Create a new Mapping called **m5_Load_ODS_PERSONNEL_STOCK_UNITS_*xx***

> **Note:** PowerCenter has many options that customize the appearance and functionality of the client applications. In this case, we want to turn off automatic creation of Source Qualifiers so we can use a single SQ to create a homogeneous join of two Source definitions.

2) From the menu, select **Tools → Options**.

3) In the Options dialogue:

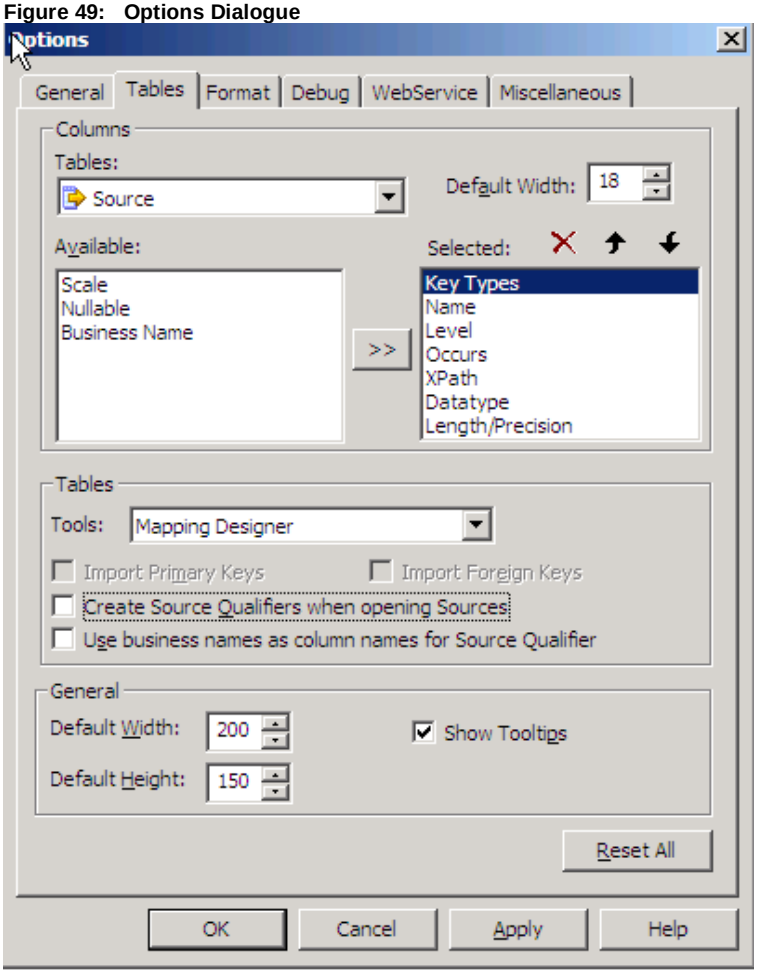
   a) Select the **Tables** tab.

   b) Make sure that the Mapping Designer tool is selected.

   c) Uncheck "Create Source Qualifier when opening Sources."

d) The Options dialogue should now look like this:

Figure 49: Options Dialogue



e) Click **OK**.

4) Drag the relational table Sources SC_STG_EMPLOYEES and SC_STG_DEALERSHIP from the Sources – STG folder onto the Mapping Designer workspace.

5) In the following steps, you will create a Source Qualifier to join the tables using the common field DEALERSHIP_ID.

Tip: Note that the fields are of the same data type – if they were not, you could not join the tables with a single Source Qualifier.

Performance Note: Extensive discussion can ensue when deciding whether it is better to have the tables joined in the database or by PowerCenter. In general, when the tables have primary keys and indexes, it is better to join them in the database.

When you are joining more than three tables, database optimizers may or may not devise a plan that leverages keys and indexes to avoid unnecessary full table scans. If a database SQL plan analysis indicates that the database is engaging in multiple full table scans, consider using PowerCenter to join at least some of the relational tables together.

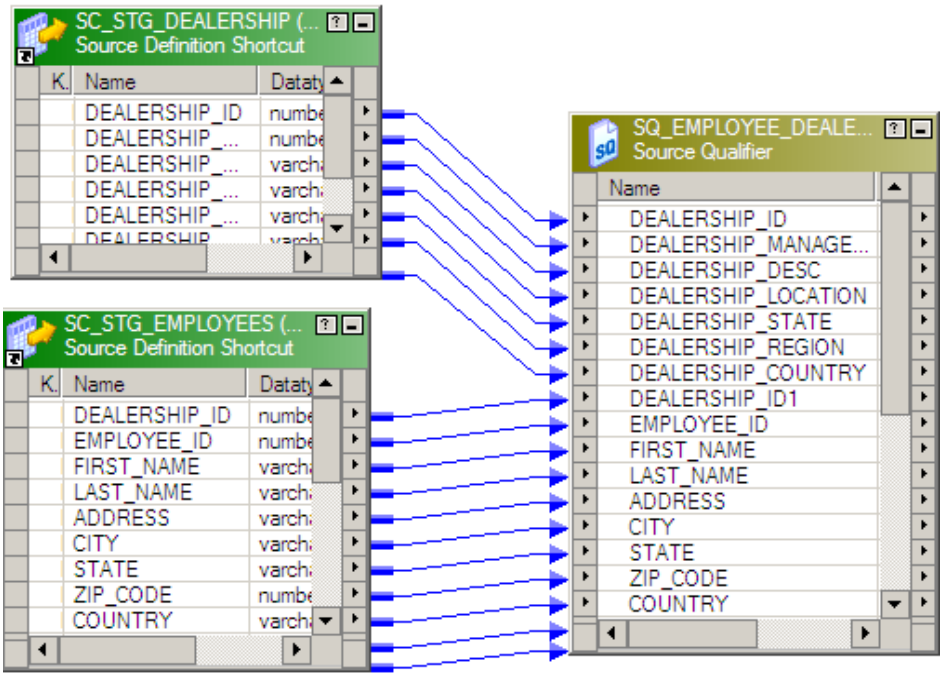
a) On the Designer toolbar, click the Source Qualifier transformation button ( ).

b) Click again on the Mapping Designer workspace.

c) In the "Select Sources for Source Qualifier Transformation" dialogue, make sure that both SC_STG_DEALERSHIP and SC_STG_EMPLOYEES are selected, then click **OK**.

d) A single Source Qualifier will be created, with all fields from both sources feeding into it.

Figure 50: Create the Source Qualifier



e) Double-click the Source Qualifier to edit it.

f) Rename it **SQ_EMPLOYEE_DEALERSHIP**.

g) Select the Properties tab.

h) In the User Defined Join field, click the bent arrow to open the SQL Editor and edit the property.

Tip: *Do not* use the "Sql Query" field for the Join condition. This will cause the workflow to fail.

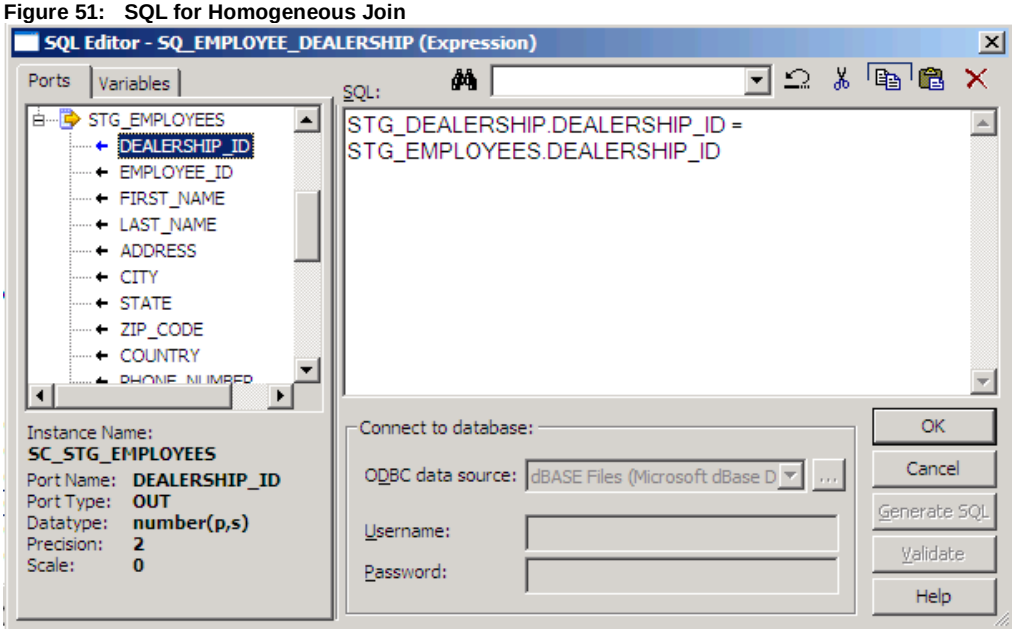
i) Under STG_DEALERSHIP, double-click DEALERSHIP_ID to place it in the SQL box.

j) Type an equal sign (=) in the SQL box.

k) Click the plus sign by STG_EMPLOYEES to expand it and double-click DEALERSHIP_ID.

l) The SQL Editor should now look like this:

Figure 51: SQL for Homogeneous Join



m) Click **OK**.

n) Click **OK** again to close the Edit Transformations dialogue.

o) Save your work.

Hint: The mapping will not validate, as it does not yet have a target object. This is OK.

Step 2. Create a Reusable Transformation

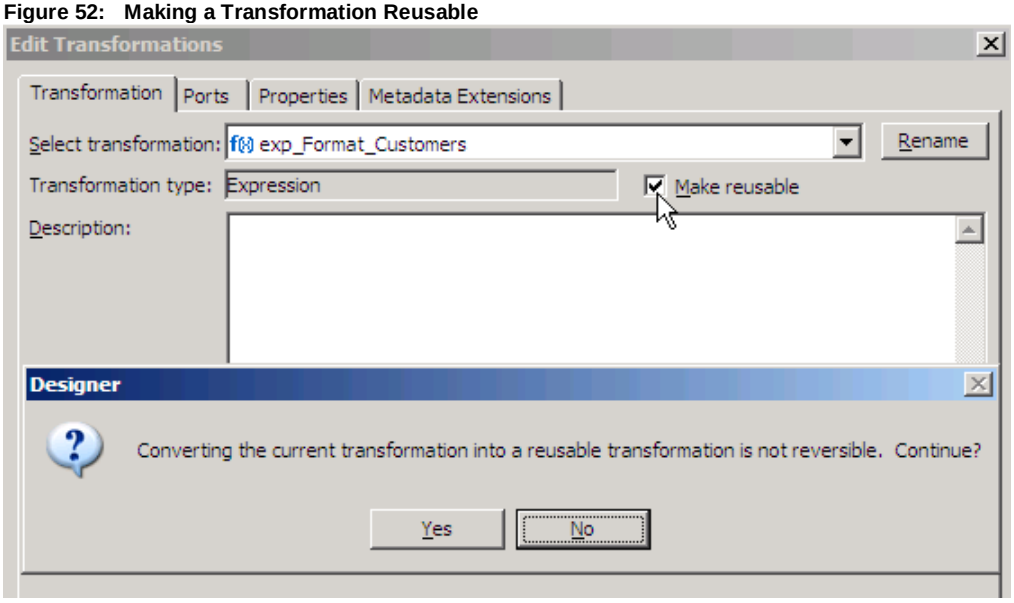
1) Open the Mapping **m4_ODS_CUSTOMERS_*xx***.

2) Double-click the transformation "exp_Format_Customers" to edit it.

   a) Make sure the Transformation tab is selected.

b) Check the option "Make Reusable."

Figure 52: Making a Transformation Reusable



Note: Making a transformation reusable is not reversible. Once done, it cannot be revoked.

Note: Also note that the best practice says that reusable objects should be created in the project shortcut folder (in this class, SC_DATA_STRUCTURES). If you, as a developer, promote an object to be reusable, you should notify your tech lead so s/he can move it to the appropriate shortcut folder. This enables the object to be properly migrated to the Test and Production environments. Once the tech lead has done this, you must modify your mapping to use the shortcut rather than the object from your local folder.

c) Click **Yes** to make the transformation reusable.

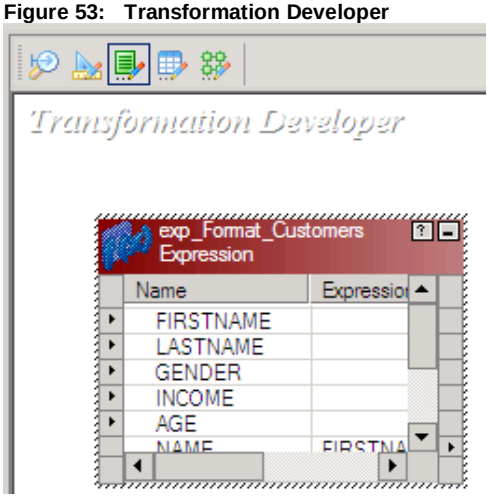
d) Select the Ports tab and note that the expressions are read-only (grayed out). You cannot edit a reusable transformation in the context of a Mapping.

e) Click **OK** to close the Edit Transformations dialogue.

3) Select the Transformation Developer tool ( ). (This can be found on the Tools tool bar).

4) In your ~Developer*xx* folder, open the Transformations subfolder and drag exp_Format_Customers onto the Transformation Developer workspace. Your screen should look like this:

Figure 53: Transformation Developer



5) Double-click the transformation to open the Edit Transformations dialogue.

   a) Select the Ports tab.

   b) Note that the expressions are editable here.

   c) Select the Transformation tab.

   d) Change the name of the transformation to **re_exp_Format_Persons** to more accurately reflect its role (formatting both Customer and Personnel data).

Note: Velocity best practice is to prefix **re_** to the name of any reusable transformation.

   e) Do not change anything else.

   f) Click **OK**.

6) Save your work.

Step 3. Complete the First Data Flow

1) Open the Mapping **m7_Load_ODS_Personnel_Stock_Units_*xx***.

2) Drag the reusable expression **re_exp_Format_Persons** from the Transformations sub-folder into the Mapping Designer workspace.

3) Connect the ports FIRST_NAME and LAST_NAME from the Source Qualifier to the ports FIRSTNAME and LASTNAME in the Expression transformation.

4) Connect the port GENDER in the Source Qualifier to the port GENDER in the Expression transformation.

5) Connect the port AGE in the Source Qualifier to the port AGE in the Expression transformation.

6) Drag the Target **SC_ODS_PERSONNEL** into the Mapping Designer workspace.

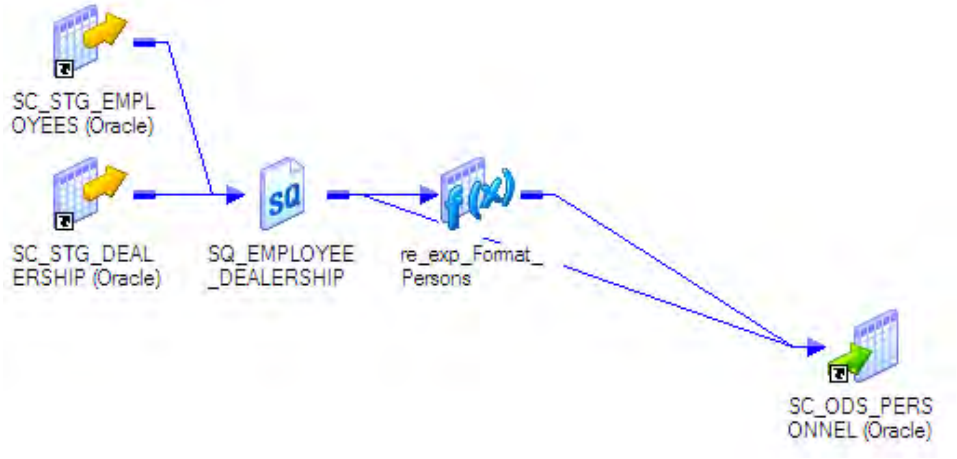
7) Connect the Target's ports according to the following table:

Expression	Port	Port in SC_ODS_PERSONNEL
re_exp_Format_Persons	NAME	NAME
	GENDER_CATEGORY	GENDER_CATEGORY
	SENIOR_FLAG	SENIOR_FLAG
SQ_Employee_Dealership	EMPLOYEE_ID	EMPLOYEE_ID
	DEALERSHIP_ID	DEALERSHIP_ID
	ZIP_CODE	POSTAL_CODE
	HIRE_DATE	HIRE_DATE
	POSITION_TYPE	POSITION_TYPE
	DEALERSHIP_MANAGER_ID	DEALERSHIP_MANAGER_ID
	DEALERSHIP_DESC	DEALERSHIP_DESCRIPTION
	DEALERSHIP_LOCATION	DEALERSHIP_LOCATION

8) Save your work. Verify that the Mapping is valid.

9) Arrange All Iconic.

Figure 54: Personnel Data Flow Arranged Iconic



Step 4. Create a Heterogeneous Join

1) Turn on automatic creation of Source Qualifiers (see Figure 1 of this lab).

2) Add the transformations to the Mapping:

   a) Drag the flatfile Source definition **SC_Inventory** and the relational Source definition **SC_STG_PRODUCT** into the Mapping Designer workspace.

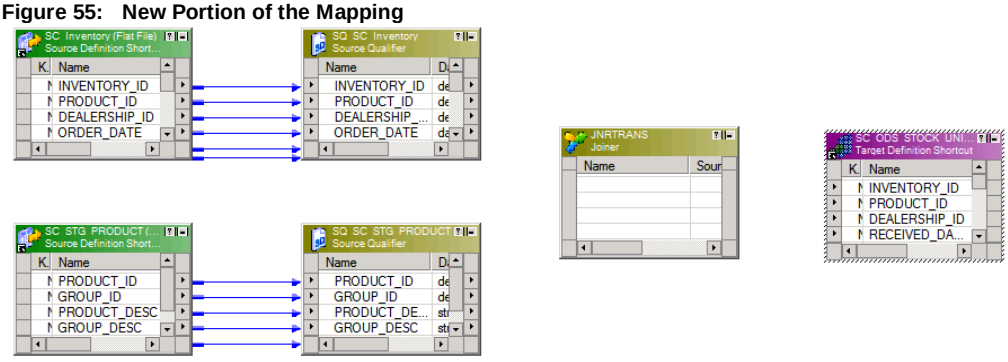
Since you will be joining a flatfile source to a relational source, you cannot use a homogeneous join here. Therefore you will use a Joiner transformation.

   b) Click the Joiner transformation icon ( ) and click again in the workspace.

   c) Drag the Target definition **SC_ODS_STOCK_UNITS** into the mapping.

d) The new portion of the Mapping should look like this:

Figure 55: New Portion of the Mapping



Performance Note: The PowerCenter Joiner transformation is fast and uses RAM rather than disk memory wherever possible. Optimizing the use of RAM can be important, particularly when RAM space is limited. Therefore, the Master side of the Joiner should be the one with the fewest duplicate keys and the fewest rows (provided this fits the logic of the join). Also, joining sorted data allows more efficient use of RAM.

In this lab, we will adhere to best practice by using STG_PRODUCT as the Master side of the Joiner. STG_PRODUCT has a much smaller number of rows than Inventory, and no duplicate keys.

3) Drag the following ports from SQ_SC_Inventory to the Joiner:

   INVENTORY_ID
   PRODUCT_ID
   DEALERSHIP_ID
   RECEIVED_DATE
   QTY_ON_HAND
   INVOICE_PRICE
   TIME_KEY
   MSRP

4) Drag the following ports from SQ_SC_STG_PRODUCT to the Joiner:

   PRODUCT_ID
   GROUP_ID
   PRODUCT_DESC
   GROUP_DESC
   DIVISION_DESC

5) Double-click the Joiner transformation to edit it.

   a) Rename it **jnr_Inventory_FF_STG_PRODUCT**.

Note: the FF is for "Flat File." As a general rule, naming conventions should be as clear as possible.

   b) Select the Ports tab.

   c) Change the name of the port PRODUCT_ID to **PRODUCT_ID_INVENTORY**.

   d) Change the name of the port PRODUCT_ID1 to **PRODUCT_ID_PRODUCT**.

   e) Select the Condition tab.

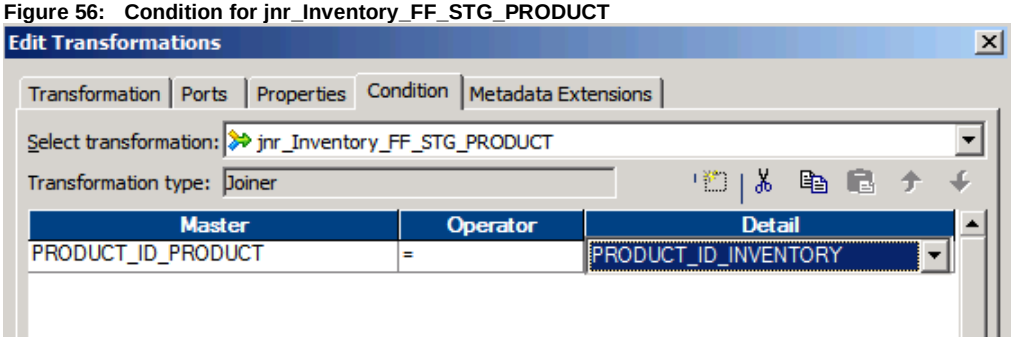
f) Create a new Join condition.

g) Make sure that the value in the Master column is PRODUCT_ID_PRODUCT and the value in the Operator column is =.

h) Change the value in the Detail column to PRODUCT_ID_INVENTORY.

i) The Condition tab should look like this:

Figure 56: Condition for jnr_Inventory_FF_STG_PRODUCT



j) Click **OK**.

Step 5. Filter Products Not In Inventory

In this step you will create a Filter transformation to remove products with no inventory.

1) Add a new Filter transformation and call it **fil_Available_Stock**.

2) Drag every port ***except*** for PRODUCT_ID_PRODUCT (which is redundant) from the Joiner transformation to the Filter transformation.

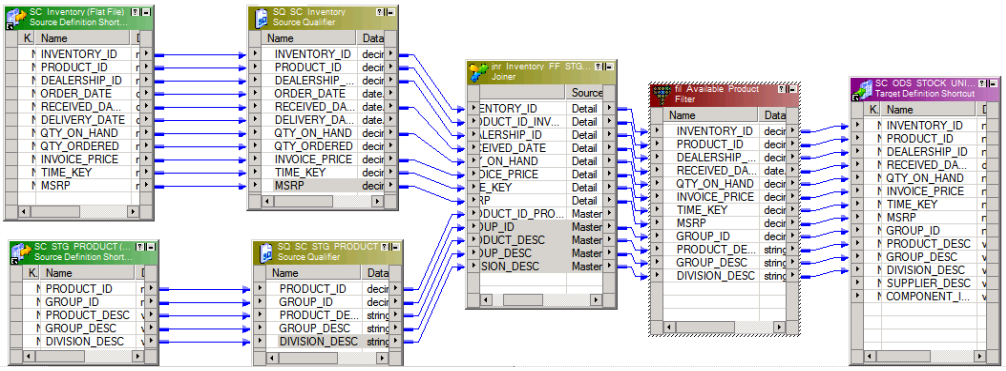
3) Set the Filter condition so that QTY_ON_HAND must be greater than 0.

4) Change the name of the PRODUCT_ID_INVENTORY port to **PRODUCT_ID**.

5) Use Autolink by Name to connect ports from the Filter transformation to the Target.

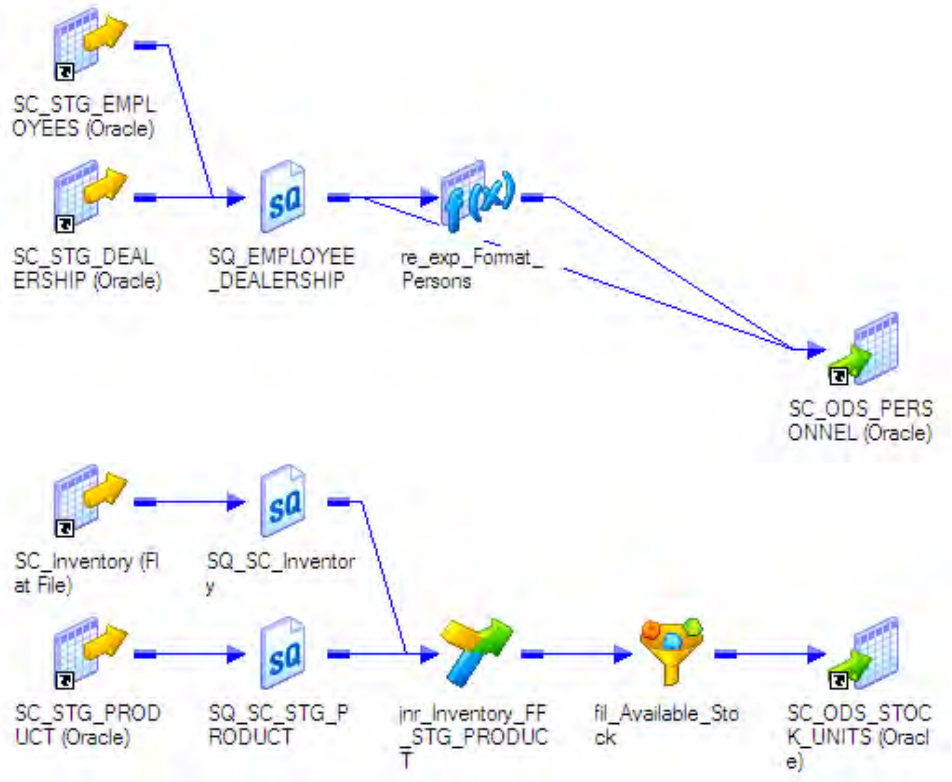
6) The data flow for products should now look like this:

Figure 57: Data flow for Products



7) Arrange All Iconic. The entire Mapping should look like this:

Figure 58: Completed Mapping m5_Load_ODS_PERSONNEL_STOCK_UNITS



8) Save your work and verify that the Mapping is valid.

Step 6. Create and Run a Workflow

1) In Workflow Manager, create a new Workflow named **wkf_Load_ODS_PERSONNEL_STOCK_*xx***.

2) Add a Session task using the mapping you just completed and link it to the Start task.

3) Edit the Session. In the Mapping tab:

   a) Set the Relational Connection for the source SQ_EMPLOYEE_DEALERSHIP to **STG*xx***.

   b) Set the Relational Connection for the source SQ_SC_STG_PRODUCT to **STG*xx***.

   c) For SQ_SC_Inventory:

      (i) The value of Source file directory should be **$PMSourceFileDir**

      (ii) The value of Source filename should be **inventory.dat**

d) For the targets SC_ODS_PERSONNEL and SC_ODS_STOCK_UNITS:

(i) Verify that the Connection is **ODS*xx***.

(ii) Set the Target load type to **Normal**.

(iii) Check the Truncate target table option.

e) Click **OK**.

4) Save your work.

5) Start the Workflow.

6) The Task Details and Source/Target Statistics for the completed Workflow should look like this:

Figure 59: Statistics for wkf_Load_ODS_PERSONNEL_STOCK_xx

s_m5_Load_ODS_PERSONNEL_STOCK_UNITS_20 [11/17/2008 6:16:06 PM]

Task Details

Attribute Name	Attribute Value
Instance Name	s_m5_Load_ODS_PERSONNEL_STOCK_UNITS_20
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	NODE01_INTDEVTEST
Start Time	11/17/2008 6:16:06 PM
End Time	11/17/2008 6:16:22 PM
Recovery Time(s)	
Status	Succeeded

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_PERS...	NODE01_I...	109	109	0	109	22781
SC_ODS_STOC...	NODE01_I...	13540	13540	0	3385	788705
SQ_EMPLOYEE...	NODE01_I...	109	109	0	109	20492
SQ_SC_Inventory	NODE01_I...	14163	14163	0	14163	1019736
SQ_SC_STG_PR...	NODE01_I...	48	48	0	48	8112

Step 7. Preview Data to Verify Results

1) Preview the data for SC_ODS_STOCK_UNITS. It should look like this:

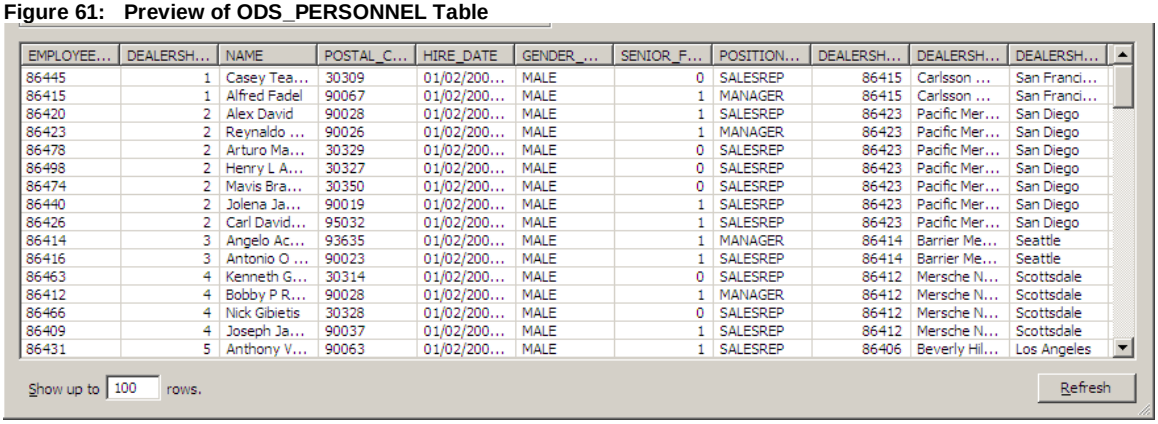
Figure 60: Preview of ODS_STOCK_UNITS Table

INVENTOR...	PRODUCT...	DEALER...	RECEIVE...	QTY_O...	INVOICE_...	TIME_KEY	MSRP	GROUP_ID	PRODUCT...	GROUP_D...	DIVISION...	SUPPLI...	CON
5022	86515	24	4/29/2002	10	1868	150147	2250	16	Careeva S...	Exterior	Accessories	NULL	NULL
5023	86537	15	9/21/2003	3	1800	173956	2250	14	TrafficPro ...	Audio and ...	Accessories	NULL	NULL
5024	86527	18	9/6/2003	3	731	170038	870	18	Car Cover	Wind and ...	Accessories	NULL	NULL
5025	86518	5	9/7/2002	5	2136	133600	2670	17	17" Sporti...	Wheels	Accessories	NULL	NULL
5026	86519	8	2/16/2003	25	2270	184007	2670	17	17" Sporti...	Wheels	Accessories	NULL	NULL
5027	86503	28	7/26/2002	3	585	100611	705	11	Ski/Snowb...	Touring	Accessories	NULL	NULL
5029	86501	7	12/31/2003	5	573	82813	690	11	Roof Box	Touring	Accessories	NULL	NULL
5031	86524	14	3/16/2003	10	779	93811	950	18	Hard Top ...	Wind and ...	Accessories	NULL	NULL
5033	86528	4	10/7/2003	6	357	170905	425	18	Leather Mi...	Wind and ...	Accessories	NULL	NULL
5034	2002	27	9/24/2002	23	43350	90955	51000	1	V6 Roadster	Sports	Vehicles	NULL	NULL
5035	86528	19	9/23/2002	5	349	81139	425	18	Leather Mi...	Wind and ...	Accessories	NULL	NULL
5036	86521	21	9/25/2003	5	3485	83125	4250	17	18" Caree...	Wheels	Accessories	NULL	NULL
5037	86503	6	5/22/2002	10	564	174955	705	11	Ski/Snowb...	Touring	Accessories	NULL	NULL
5041	86524	12	8/7/2003	8	760	110435	950	18	Hard Top ...	Wind and ...	Accessories	NULL	NULL
5042	86525	16	5/24/2002	10	183	123947	220	18	Wind defle...	Wind and ...	Accessories	NULL	NULL

Show up to 100 rows. Refresh

2) Preview the data for ODS_PERSONNEL. It should look like this:

Figure 61: Preview of ODS_PERSONNEL Table



EMPLOYEE...	DEALERSH...	NAME	POSTAL_C...	HIRE_DATE	GENDER_...	SENIOR_F...	POSITION...	DEALERSH...	DEALERSH...	DEALERSH...
86445	1	Casey Tea...	30309	01/02/200...	MALE	0	SALESREP	86415	Carlsson ...	San Franci...
86415	1	Alfred Fadel	90067	01/02/200...	MALE	1	MANAGER	86415	Carlsson ...	San Franci...
86420	2	Alex David	90028	01/02/200...	MALE	1	SALESREP	86423	Pacific Mer...	San Diego
86423	2	Reynaldo ...	90026	01/02/200...	MALE	1	MANAGER	86423	Pacific Mer...	San Diego
86478	2	Arturo Ma...	30329	01/02/200...	MALE	0	SALESREP	86423	Pacific Mer...	San Diego
86498	2	Henry L A...	30327	01/02/200...	MALE	0	SALESREP	86423	Pacific Mer...	San Diego
86474	2	Mavis Bra...	30350	01/02/200...	MALE	0	SALESREP	86423	Pacific Mer...	San Diego
86440	2	Jolena Ja...	90019	01/02/200...	MALE	1	SALESREP	86423	Pacific Mer...	San Diego
86426	2	Carl David...	95032	01/02/200...	MALE	1	SALESREP	86423	Pacific Mer...	San Diego
86414	3	Angelo Ac...	93635	01/02/200...	MALE	1	MANAGER	86414	Barrier Me...	Seattle
86416	3	Antonio O ...	90023	01/02/200...	MALE	1	SALESREP	86414	Barrier Me...	Seattle
86463	4	Kenneth G...	30314	01/02/200...	MALE	0	SALESREP	86412	Mersche N...	Scottsdale
86412	4	Bobby P R...	90028	01/02/200...	MALE	1	MANAGER	86412	Mersche N...	Scottsdale
86466	4	Nick Gibietis	30328	01/02/200...	MALE	0	SALESREP	86412	Mersche N...	Scottsdale
86409	4	Joseph Ja...	90037	01/02/200...	MALE	1	SALESREP	86412	Mersche N...	Scottsdale
86431	5	Anthony V...	90063	01/02/200...	MALE	1	SALESREP	86406	Beverly Hil...	Los Angeles

Show up to 100 rows. Refresh

Extra Credit Lab:

Note that there are entries in ODS_PERSONNEL with null Employee IDs. Modify the part of the Mapping that populates this table to substitute the default value 99999.

Hint: Use the Default Value property in the Expression transformation.

Lab 6-1: Using the Lookup Transformation

Scenario:

- The ODS database needs a table that calculates such sales values as gross and net profit.
- The sales executives want to know if more cars are sold on some weekdays than others. Neither the sales date nor the name of the day of the week is in the Transaction table. They must be added to the Sales table in the ODS.
- Calculating gross and net profit is simplified by calculating an intermediate metric, Margin.

Goals:

- Use a Lookup transformation to import dates
- Use a variable to calculate net and gross profit

Duration:

40 minutes

Instructions

Step 8. Create and Configure a Lookup Transformation

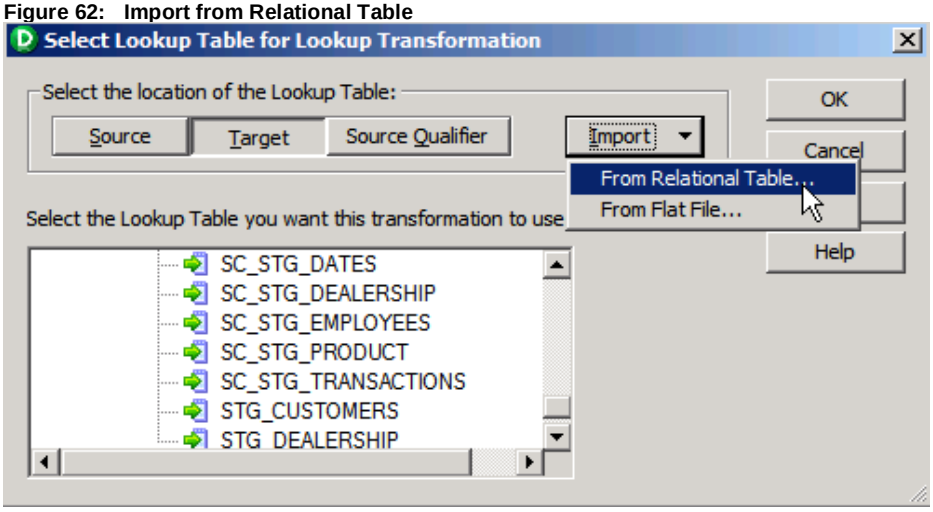
1) In PowerCenter Designer, create a new Mapping named **m6_Load_ODS_SALES_*xx***.

   a) Drag in the source **SC_STG_TRANSACTIONS** and the target **SC_ODS_SALES**

2) Click the Lookup transformation icon ( ), then click again in the Mapping Designer workspace.

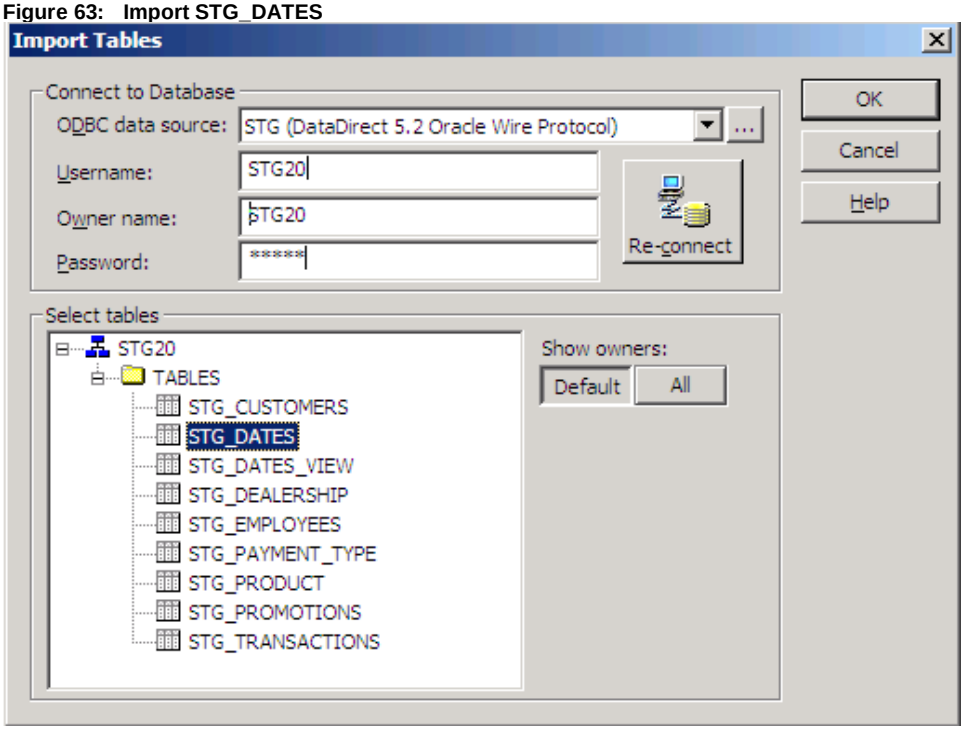
   a) Click the Import button and select **From Relational Table**

Figure 62: Import from Relational Table



b) In the Import Tables dialogue, log into the **STG*xx*** schema and select the STG_DATES table.

Figure 63: Import STG_DATES



3) Drag the port DATE_ID from the Source Qualifier and drop it on the Lookup transformation to create a link.

4) Double-click the Lookup transformation to edit it.

   a) Rename the transformation **lkp_STG_DATES_WEEKDAY**.

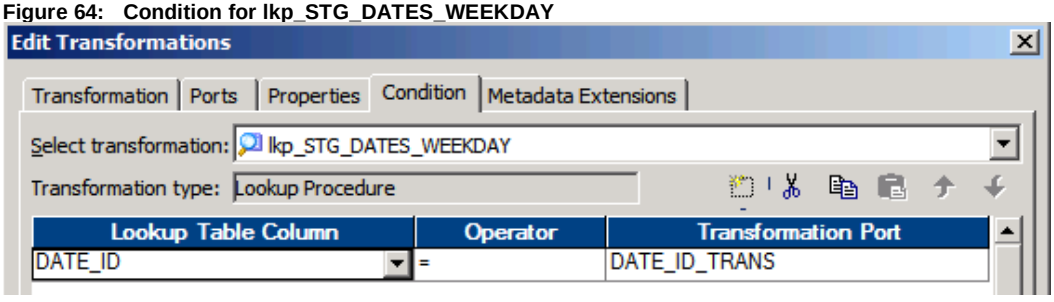
   b) Click the Ports tab.

   c) Rename the port DATE_ID1 to **DATE_ID_TRANS** (to indicate it comes from the transactions table).

   d) Click the Condition tab.

   e) Create a new condition. It should appear as shown.

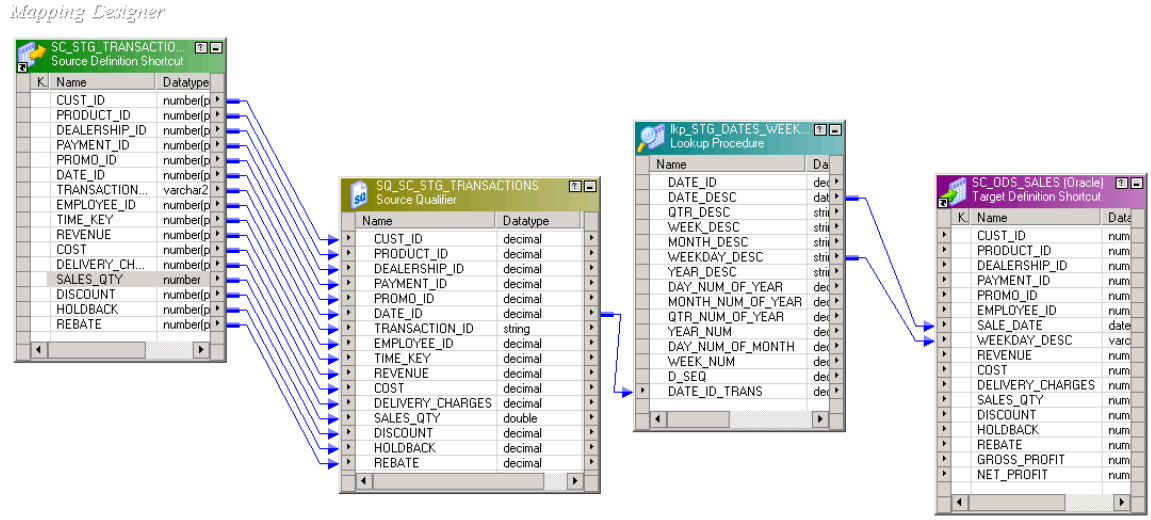
Figure 64: Condition for lkp_STG_DATES_WEEKDAY



   f) Click **OK** to close the Edit Transformations dialogue.

5) Connect the ports DATE_DESC and WEEKDAY_DESC to the Target ports SALE_DATE and WEEKDAY_DESC respectively. The Mapping should now look like this:

Figure 65: Mapping with Lookup Configured



6) Iconize the Lookup transformation and drag it out of the way.

7) Save your work.

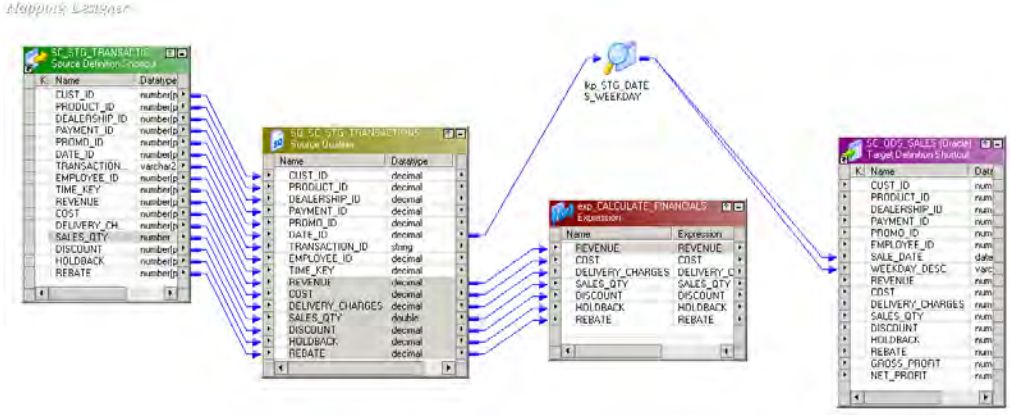
Step 9. Create an Expression using a Variable Port

1) Create an Expression transformation named **exp_CALCULATE_FINANCIALS**.

2) Drag the following ports from the Source Qualifier to the Expression transformation:
REVENUE
COST
DELIVERY_CHARGES
SALES_QTY
DISCOUNT
HOLDBACK
REBATE

   a) The mapping should now look like this:

Figure 66: Mapping with Expression Added



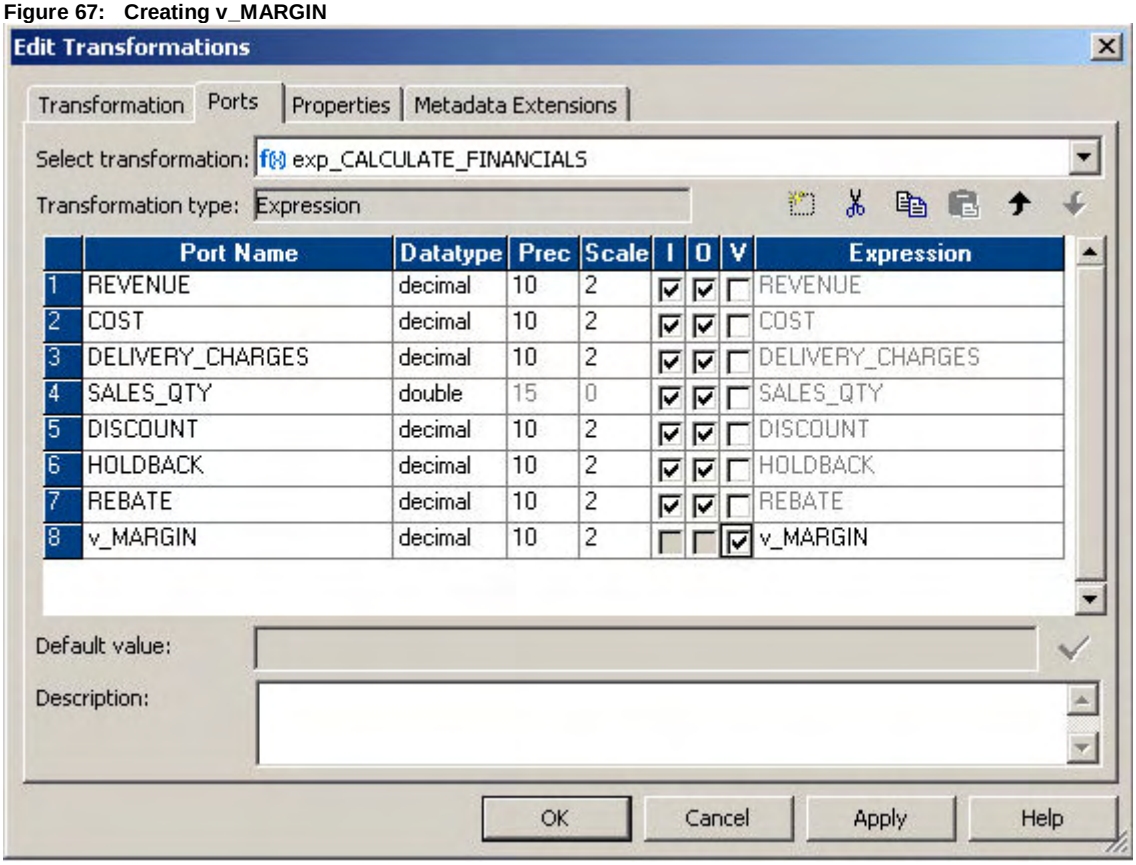
3) Edit exp_CALCULATE_FINANCIALS and select the Ports tab.

a) Create a new port called **v_MARGIN**, with Datatype decimal 10.2 and port type Variable.

Note: the "v" indicates that this port is a variable.

(i) It should look like this:

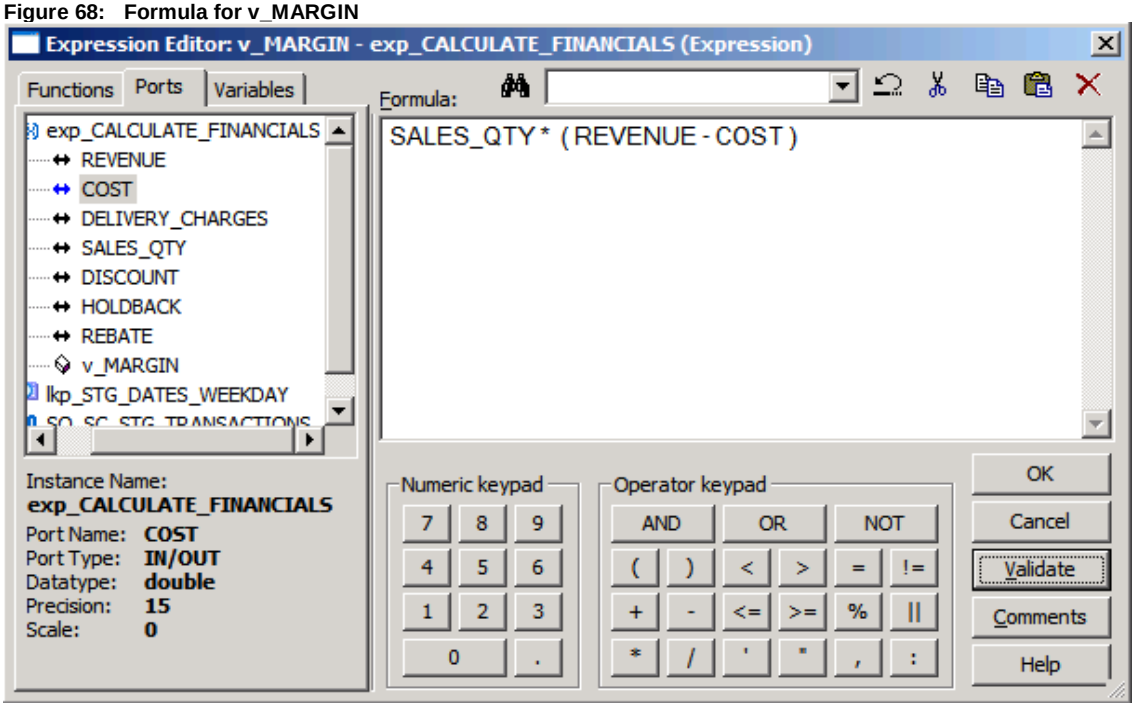
Figure 67: Creating v_MARGIN



(ii) Open the Expression Editor for v_MARGIN.

(iii) Margin is calculated by multiplying SALES_QTY by the difference of REVENUE and COST, as shown.

Figure 68: Formula for v_MARGIN



b) Create another new port named **GROSS_PROFIT** with Datatype decimal 10.2 and set it to Output only.

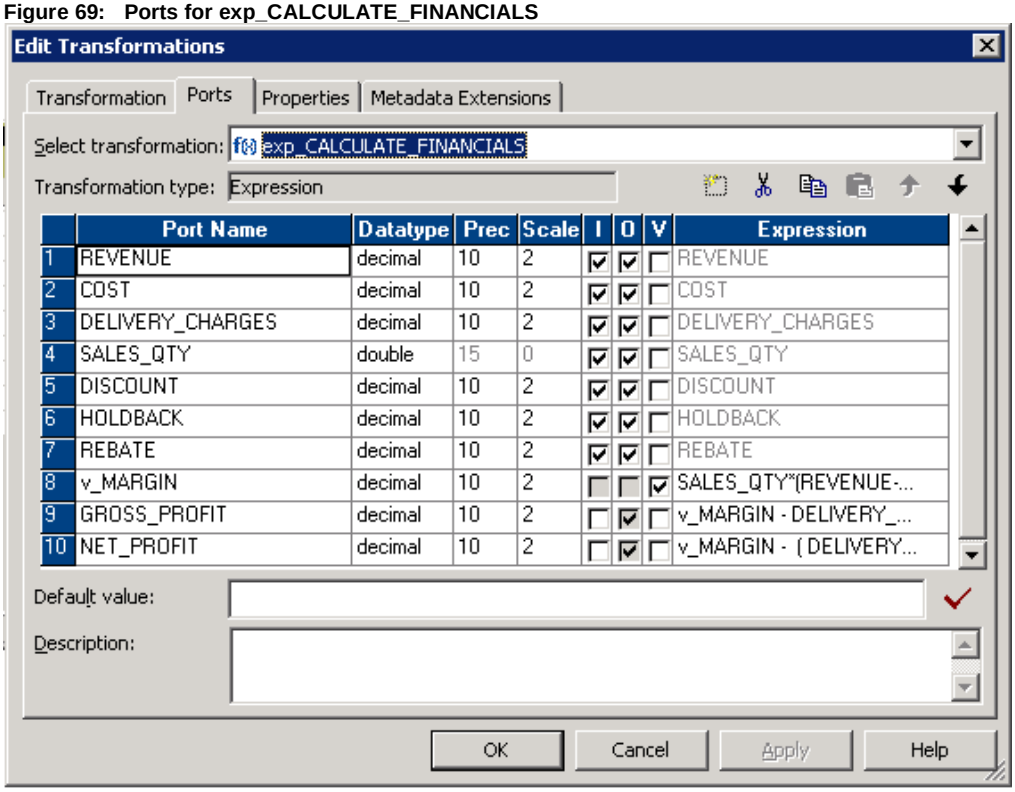
(i) Using the Expression Editor, set its formula to v_**MARGIN** – **DELIVERY_CHARGES**.

c) Create another new port named **NET_PROFIT** with Datatype decimal 10.2 and set it to Output only.

(i) Using the Expression Editor, set its formula to **v_MARGIN**– (**DELIVERY_CHARGES + DISCOUNT + HOLDBACK + REBATE**)

d) The Ports tab should look like this:

Figure 69: Ports for exp_CALCULATE_FINANCIALS



e) Click **OK**

f) Save your work.

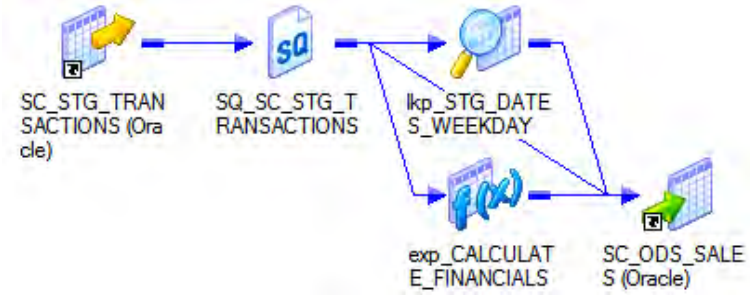
Step 10. Finish the Mapping

1) Connect the ports GROSS_PROFIT and NET_PROFIT from the Expression transformation to the ports with the same names in the Target definition.

2) Autolink by name from the Source Qualifier to the Target definition.

3) Save your work and ensure that the Mapping is valid.

Figure 70: M6_Load_ODS_SALES



Step 11. Create and Run the Workflow

1) In Workflow Manager, create a Workflow called **wkf_Load_ODS_SALES_*xx***.
2) Add a Session task to the Workflow based on the Mapping **m6_Load_ODS_SALES_*xx***.
3) Link the Start task to the Session task
4) Edit the Session task. In the Mapping tab:
   a) The Source table needs the STG*xx* relational connection
   b) The target table needs the ODS*xx* relational connection.
      (i) Set the load type to normal
      (ii) Select the Truncate table option
   c) Click the object lkp_STG_DATES_WEEKDAY
      (i) The Type of the connection should already be set to Relational.
         (a) If not, set it.
      (ii) Set the value of the Relational connection to STG*xx*.
5) Click **OK**.
6) Save your work.
7) Run the workflow.

Figure 71: Task Details and Source/Target Statistics for wkf_Load_ODS_SALES

s_m6_Load_ODS_SALES_20 [11/17/2008 6:33:30 PM]

Task Details

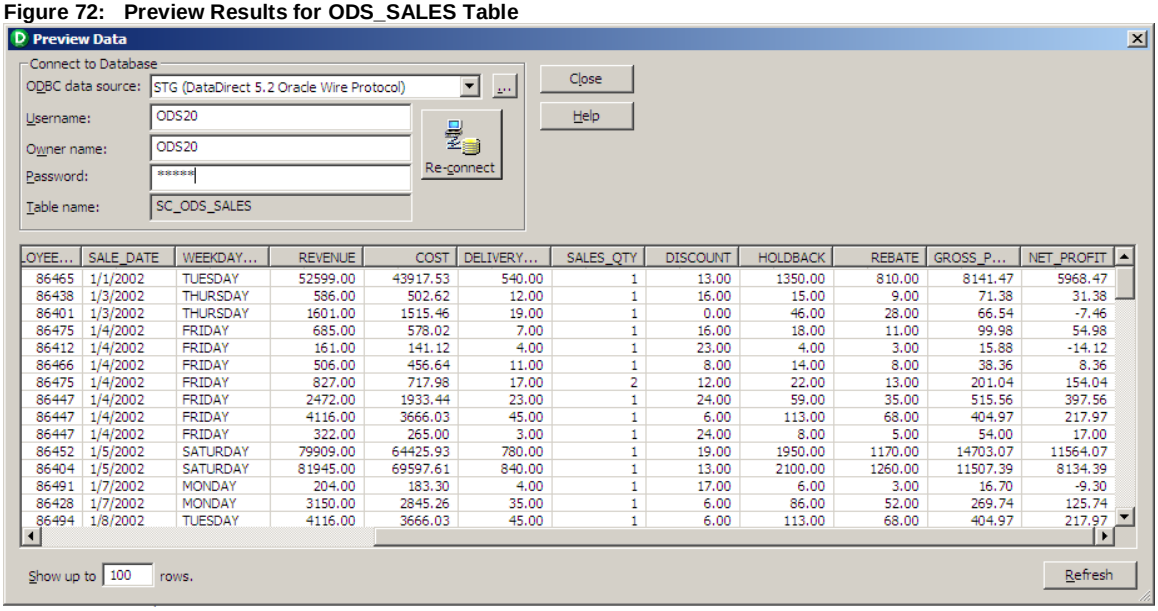
Attribute Name	Attribute Value
Instance Name	s_m6_Load_ODS_SALES_20
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	NODE01_INTDEVTEST
Start Time	11/17/2008 6:33:30 PM
End Time	11/17/2008 6:33:41 PM
Recovery Time(s)	
Status	Succeeded

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_SALES	NODE01_I...	5475	5475	0	5475	804825
SQ_SC_STG_TR...	NODE01_I...	5475	5475	0	5475	613200

8) Preview the data for the ODS_SALES table.

Figure 72: Preview Results for ODS_SALES Table



OYEE...	SALE_DATE	WEEKDAY...	REVENUE	COST	DELIVERY...	SALES_QTY	DISCOUNT	HOLDBACK	REBATE	GROSS_P...	NET_PROFIT
86465	1/1/2002	TUESDAY	52599.00	43917.53	540.00	1	13.00	1350.00	810.00	8141.47	5968.47
86438	1/3/2002	THURSDAY	586.00	502.62	12.00	1	16.00	15.00	9.00	71.38	31.38
86401	1/3/2002	THURSDAY	1601.00	1515.46	19.00	1	0.00	46.00	28.00	66.54	-7.46
86475	1/4/2002	FRIDAY	685.00	578.02	7.00	1	16.00	18.00	11.00	99.98	54.98
86412	1/4/2002	FRIDAY	161.00	141.12	4.00	1	23.00	4.00	3.00	15.88	-14.12
86466	1/4/2002	FRIDAY	506.00	456.64	11.00	1	8.00	14.00	8.00	38.36	8.36
86475	1/4/2002	FRIDAY	827.00	717.98	17.00	2	12.00	22.00	13.00	201.04	154.04
86447	1/4/2002	FRIDAY	2472.00	1933.44	23.00	1	24.00	59.00	35.00	515.56	397.56
86447	1/4/2002	FRIDAY	4116.00	3666.03	45.00	1	6.00	113.00	68.00	404.97	217.97
86447	1/4/2002	FRIDAY	322.00	265.00	3.00	1	24.00	8.00	5.00	54.00	17.00
86452	1/5/2002	SATURDAY	79909.00	64425.93	780.00	1	19.00	1950.00	1170.00	14703.07	11564.07
86404	1/5/2002	SATURDAY	81945.00	69597.61	840.00	1	13.00	2100.00	1260.00	11507.39	8134.39
86491	1/7/2002	MONDAY	204.00	183.30	4.00	1	17.00	6.00	3.00	16.70	-9.30
86428	1/7/2002	MONDAY	3150.00	2845.26	35.00	1	6.00	86.00	52.00	269.74	125.74
86494	1/8/2002	TUESDAY	4116.00	3666.03	45.00	1	6.00	113.00	68.00	404.97	217.97

Lab 7-1: Using Aggregators and Sorters

Scenario:

- In addition to the ODS_Sales table, sales executives need tables that summarize sales at the weekly and monthly level. This will facilitate report generation and ad-hoc analysis.

Goals:

- Use a Lookup transformation to add week and month information to the data rows
- Split the data stream to feed two Aggregator transformations and write data out to two separate tables
- Use Sorter transformations to improve efficiency of the mapping

Duration:

75 minutes

Instructions

Step 1. Create the Mapping

1) Create a new Mapping named **m7_Sales_Summaries_*xx***.
2) Drag in the Source definition **SC_ODS_SALES** and the Target definitions **SC_ODS_SALES_BY_WEEK** and **SC_ODS_SALES_BY_MONTH**.
3) Add a Lookup transformation to the Mapping.

Note: If you need help with this step, consult the instructions for Lab 6-1.

   a) Import the table STG_DATES as its source.
   b) Drag the port SALE_DATE from the Source Qualifier to the Lookup transformation.
   c) Edit the Lookup transformation.
      (i) Change its name to **lkp_STG_DATES_WEEK_MONTH**.
      (ii) Set its Lookup condition to **DATE_DESC = SALE_DATE**

Step 2. Create and Configure an Aggregator to Summarize Data by Month

1) Use the Aggregator icon () to add an Aggregator transformation to the Mapping.

2) Drag the following ports to the Aggregator:

   a) From lkp_STG_DATES_WEEK_MONTH:
      MONTH_DESC

   b) From the Source Qualifier SQ_SC_ODS_SALES:

REVENUE	HOLDBACK
COST	REBATE
DELIVERY_CHARGE	GROSS_PROFIT
SALES_QTY	NET_PROFIT
DISCOUNT	

   c) From the Target SC_ODS_SALES_BY_MONTH:

REVENUE	HOLDBACK
COST	REBATE
DELIVERY_CHARGE	GROSS_PROFIT
SALES_QTY	NET_PROFIT
DISCOUNT	

Note: PowerCenter will automatically rename these ports with a numerical 1 at the end of each port name to avoid having duplicate port names.

3) Edit the Aggregator.

   a) Rename it **agg_SUM_BY_MONTH**.

   b) Select the Ports tab.

   c) Check the "Group By" checkbox for the port MONTH_DESC

   d) For each of the ports imported from SC_ODS_SALES_BY_MONTH:

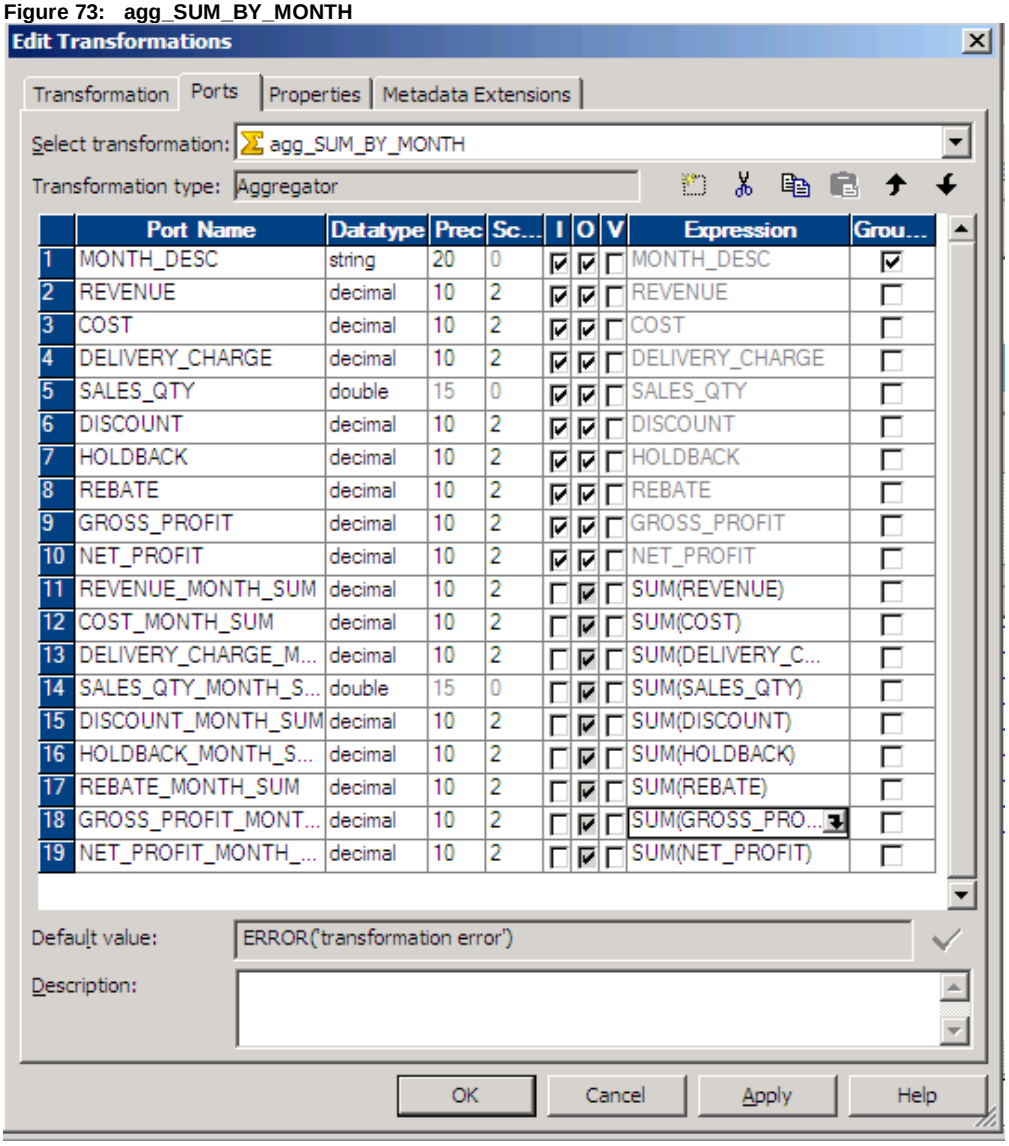
      (i) Change the numerical 1 in the name to _MONTH_SUM (so that, for example, REVENUE1 becomes REVENUE_MONTH_SUM and so on).

      (ii) Change the port type from input only to output only.

      (iii) Edit each of the expressions for the _MONTH_SUM ports to calculate a sum. For example, the expression for REVENUE_MONTH_SUM should be SUM(REVENUE).

e) The Aggregator should now look like this:

Figure 73: agg_SUM_BY_MONTH



	Port Name	Datatype	Prec	Sc...	I	O	V	Expression	Grou...
1	MONTH_DESC	string	20	0	☑	☑	☐	MONTH_DESC	☑
2	REVENUE	decimal	10	2	☑	☑	☐	REVENUE	☐
3	COST	decimal	10	2	☑	☑	☐	COST	☐
4	DELIVERY_CHARGE	decimal	10	2	☑	☑	☐	DELIVERY_CHARGE	☐
5	SALES_QTY	double	15	0	☑	☑	☐	SALES_QTY	☐
6	DISCOUNT	decimal	10	2	☑	☑	☐	DISCOUNT	☐
7	HOLDBACK	decimal	10	2	☑	☑	☐	HOLDBACK	☐
8	REBATE	decimal	10	2	☑	☑	☐	REBATE	☐
9	GROSS_PROFIT	decimal	10	2	☑	☑	☐	GROSS_PROFIT	☐
10	NET_PROFIT	decimal	10	2	☑	☑	☐	NET_PROFIT	☐
11	REVENUE_MONTH_SUM	decimal	10	2	☐	☑	☐	SUM(REVENUE)	☐
12	COST_MONTH_SUM	decimal	10	2	☐	☑	☐	SUM(COST)	☐
13	DELIVERY_CHARGE_M...	decimal	10	2	☐	☑	☐	SUM(DELIVERY_C...	☐
14	SALES_QTY_MONTH_S...	double	15	0	☐	☑	☐	SUM(SALES_QTY)	☐
15	DISCOUNT_MONTH_SUM	decimal	10	2	☐	☑	☐	SUM(DISCOUNT)	☐
16	HOLDBACK_MONTH_S...	decimal	10	2	☐	☑	☐	SUM(HOLDBACK)	☐
17	REBATE_MONTH_SUM	decimal	10	2	☐	☑	☐	SUM(REBATE)	☐
18	GROSS_PROFIT_MONT...	decimal	10	2	☐	☑	☐	SUM(GROSS_PRO...	☐
19	NET_PROFIT_MONTH_...	decimal	10	2	☐	☑	☐	SUM(NET_PROFIT)	☐

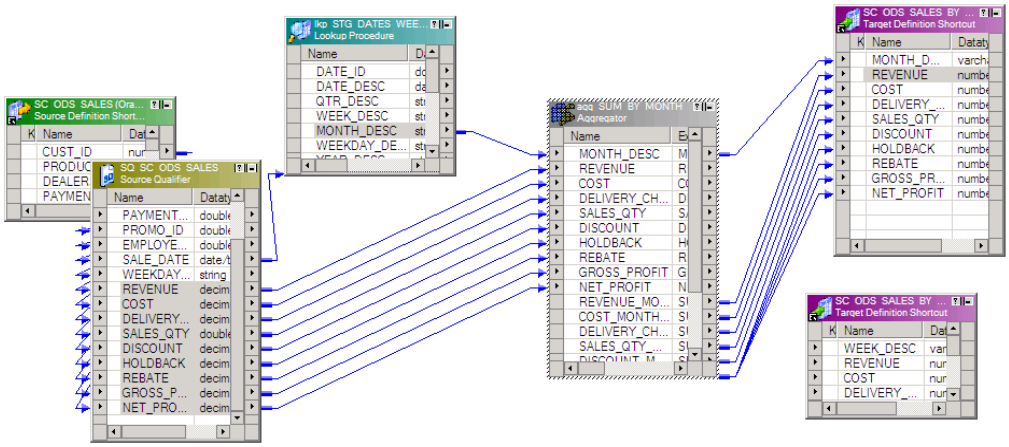
4) Click **OK** to exit the Edit Transformations dialogue.

5) Link the MONTH_DESC port and the MONTH_SUM ports to the appropriate ports in the Target SC_ODS_SALES_BY_MONTH.

6) Save the Mapping. It may not validate at this time.

7) The Mapping should look like this:

Figure 74: Mapping with One Aggregator Added



Step 3. Create and Configure an Aggregator to Summarize Data by Week

1) Copy the Aggregator:

   a) Right-click the Aggregator and select "Copy"

   b) Right-click in the workspace and select "Paste"

2) Drag the following ports to the new Aggregator:

   a) Connect WEEK_DESC from **lkp_STG_SALES_WEEK_MONTH** to the MONTH_DESC port of the new Aggregator

   b) Connect the following ports from the Source Qualifier **SQ_SC_ODS_SALES** to the new Aggregator:

REVENUE	HOLDBACK
COST	REBATE
DELIVERY_CHARGE	GROSS_PROFIT
SALES_QTY	NET_PROFIT
DISCOUNT	

3) Edit the new Aggregator.

   a) Rename it **agg_SUM_BY_WEEK**.

   b) Select the Ports tab.

   c) Change the name of the port MONTH_DESC to WEEK_DESC

   d) For each of the MONTH_SUM ports:

      (i) Change the name to _WEEK_SUM (so that, for example, REVENUE_MONTH_SUM becomes REVENUE_WEEK_SUM and so on)

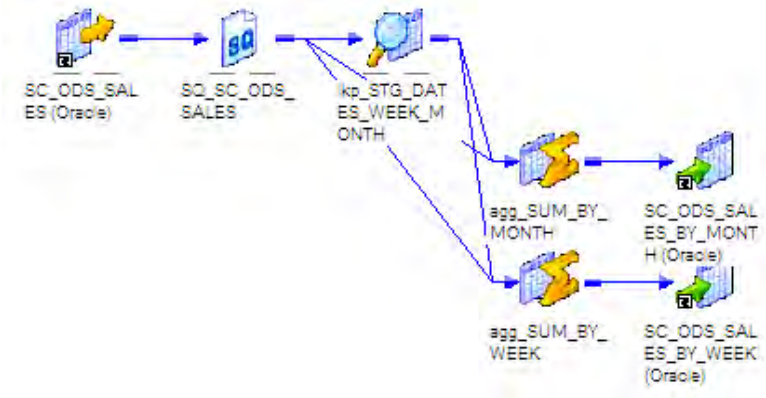
4) Click **OK** to exit the Edit Transformations dialogue.

5) Connect the WEEK_DESC and WEEK_SUM ports of the new Aggregator to their counterparts in the target **SC_ODS_SALES_BY_WEEK**.

6) Save the Mapping and make sure that it validates.

7) The Mapping should look like this when you Arrange All Iconic:

Figure 75: M7_Sales_Summaries



Step 4. Create and Run the Workflow

1) Create a new Workflow named **wkf_Load_Sales_Summaries_*xx***

2) The Session should both read from and write to Relational connection ODS*xx*.

3) Set the relational connection for the Lookup to STG*xx*.

4) Be sure to set the Load type to Normal and the Truncate table option on (checked).

5) Save and run the Workflow.

Figure 76: Task Details and Source/Target Statistics for s_m9_Sales_Summaries

s_m7_Sales_Summaries_20 [11/17/2008 7:09:10 PM]

Task Details

Attribute Name	Attribute Value
Instance Name	s_m7_Sales_Summaries_20
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	NODE01_INTDEVTEST
Start Time	11/17/2008 7:09:10 PM

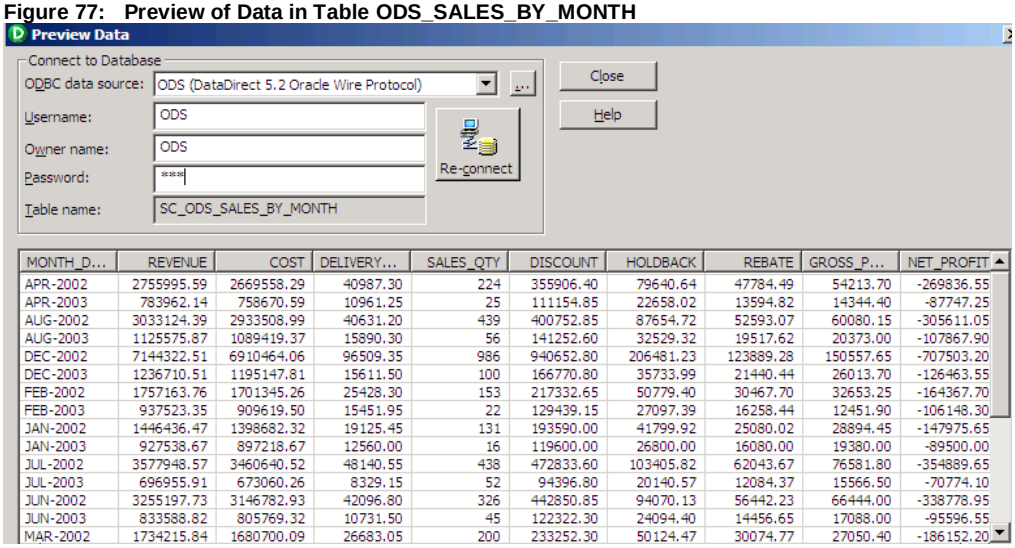
Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_SALES...	NODE01_I...	24	24	0	24	2232
SC_ODS_SALES...	NODE01_I...	106	106	0	106	9858
SQ_SC_ODS_SA...	NODE01_I...	5475	5475	0	5475	481800

a) Note how long the Session task took to run:

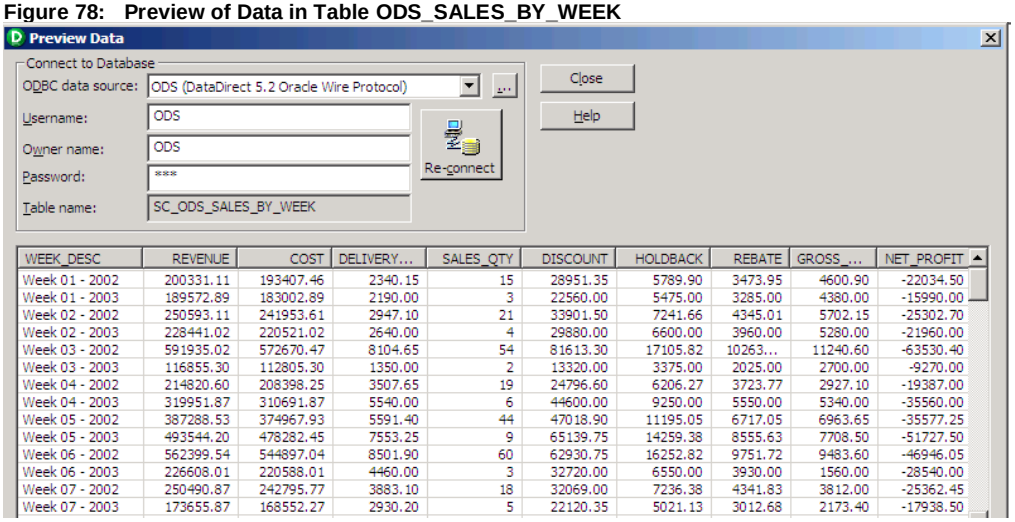
_____

Figure 77: Preview of Data in Table ODS_SALES_BY_MONTH



MONTH_D...	REVENUE	COST	DELIVERY...	SALES_QTY	DISCOUNT	HOLDBACK	REBATE	GROSS_P...	NET_PROFIT
APR-2002	2755995.59	2669558.29	40987.30	224	355906.40	79640.64	47784.49	54213.70	-269836.55
APR-2003	783962.14	758670.59	10961.25	25	111154.85	22658.02	13594.82	14344.40	-87747.25
AUG-2002	3033124.39	2933508.99	40631.20	439	400752.85	87654.72	52593.07	60080.15	-305611.05
AUG-2003	1125575.87	1089419.37	15890.30	56	141252.60	32529.32	19517.62	20373.00	-107867.90
DEC-2002	7144322.51	6910464.06	96509.35	986	940652.80	206481.23	123889.28	150557.65	-707503.20
DEC-2003	1236710.51	1195147.81	15611.50	100	166770.80	35733.99	21440.44	26013.70	-126463.55
FEB-2002	1757163.76	1701345.26	25428.30	153	217332.65	50779.40	30467.70	32653.25	-164367.70
FEB-2003	937523.35	909619.50	15451.95	22	129439.15	27097.39	16258.44	12451.90	-106148.30
JAN-2002	1446436.47	1398682.32	19125.45	131	193590.00	41799.92	25080.02	28894.45	-147975.65
JAN-2003	927538.67	897218.67	12560.00	16	119600.00	26800.00	16080.00	19380.00	-89500.00
JUL-2002	3577948.57	3460640.52	48140.55	438	472833.60	103405.82	62043.67	76581.80	-354889.65
JUL-2003	696955.91	673060.26	8329.15	52	94396.80	20140.57	12084.37	15566.50	-70774.10
JUN-2002	3255197.73	3146782.93	42096.80	326	442850.85	94070.13	56442.23	66444.00	-338778.95
JUN-2003	833588.82	805769.32	10731.50	45	122322.30	24094.40	14456.65	17088.00	-95596.55
MAR-2002	1734215.84	1680700.09	26683.05	200	233252.30	50124.47	30074.77	27050.40	-186152.20

Figure 78: Preview of Data in Table ODS_SALES_BY_WEEK



WEEK_DESC	REVENUE	COST	DELIVERY...	SALES_QTY	DISCOUNT	HOLDBACK	REBATE	GROSS_...	NET_PROFIT
Week 01 - 2002	200331.11	193407.46	2340.15	15	28951.35	5789.90	3473.95	4600.90	-22034.50
Week 01 - 2003	189572.89	183002.89	2190.00	3	22560.00	5475.00	3285.00	4380.00	-15990.00
Week 02 - 2002	250593.11	241953.61	2947.10	21	33901.50	7241.66	4345.01	5702.15	-25302.70
Week 02 - 2003	228441.02	220521.02	2640.00	4	29880.00	6600.00	3960.00	5280.00	-21960.00
Week 03 - 2002	591935.02	572670.47	8104.65	54	81613.30	17105.82	10263...	11240.60	-63530.40
Week 03 - 2003	116855.30	112805.30	1350.00	2	13320.00	3375.00	2025.00	2700.00	-9270.00
Week 04 - 2002	214820.60	208398.25	3507.65	19	24796.60	6206.27	3723.77	2927.10	-19387.00
Week 04 - 2003	319951.87	310691.87	5540.00	6	44600.00	9250.00	5550.00	5340.00	-35560.00
Week 05 - 2002	387288.53	374967.93	5591.40	44	47018.90	11195.05	6717.05	6963.65	-35577.25
Week 05 - 2003	493544.20	478282.45	7553.25	9	65139.75	14259.38	8555.63	7708.50	-51727.50
Week 06 - 2002	562399.54	544897.04	8501.90	60	62930.75	16252.82	9751.72	9483.60	-46946.05
Week 06 - 2003	226608.01	220588.01	4460.00	3	32720.00	6550.00	3930.00	1560.00	-28540.00
Week 07 - 2002	250490.87	242795.77	3883.10	18	32069.00	7236.38	4341.83	3812.00	-25362.45
Week 07 - 2003	173655.87	168552.27	2930.20	5	22120.35	5021.13	3012.68	2173.40	-17938.50

Note: The basic functionality of the Mapping is complete. However, in the Production environment, where there will be millions of records, the Aggregator transformations may run very slowly.

By default, Aggregator transformations work by creating a "bucket" for each unique value in the Group By port(s). If the number of unique values is large, a great deal of memory may be dedicated to maintaining these "buckets," or the system may have to cache buckets to disk. In either case this can have a performance impact.

To prevent this, you can sort the data prior to its reaching the Aggregator. If the data is sorted on the Group By port, and the Aggregator transformation is "told" that this is the case, then there is no need to maintain many "buckets," and performance is improved.

Step 5. Add Sorters to the Mapping

1) Use the Sorter icon ( ) to add a Sorter transformation to the mapping.

2) Drag the following ports into it:

   a) From lkp_STG_SALES_WEEK_MONTH:
   MONTH_DESC

   b) From SQ_SC_ODS_SALES:

REVENUE	HOLDBACK
COST	REBATE
DELIVERY_CHARGE	GROSS_PROFIT
SALES_QTY	NET_PROFIT
DISCOUNT	

3) Edit the Sorter transformation.

   a) Rename it **srt_MONTH_DESC**

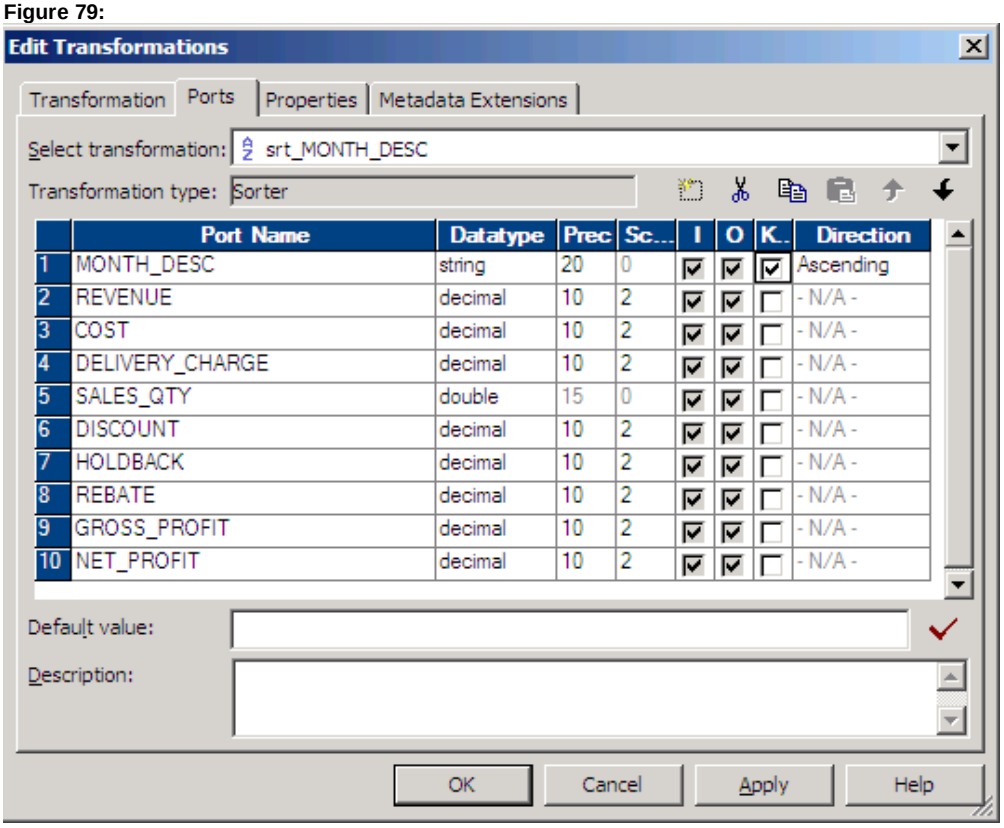
   b) Select the Ports tab.

      (i) Set the port MONTH_DESC as the Key to sort on.

      (ii) Accept the default "Ascending" sort.

   c) The Sorter should now look like this:

Figure 79:



   d) Click **OK** to close the Edit Transformations dialogue.

4) Remove all links from SQ_SC_ODS_SALES and lkp_STG_DATES_WEEK_MONTH to the agg_SUM_BY_MONTH.

5) Link the ports from srt_MONTH_DESC to agg_SUM_BY_MONTH.

6) Edit agg_SUM_BY_MONTH.

a) Select the Properties tab.

b) Check the box for "Sorted Input"

c) Click **OK**.

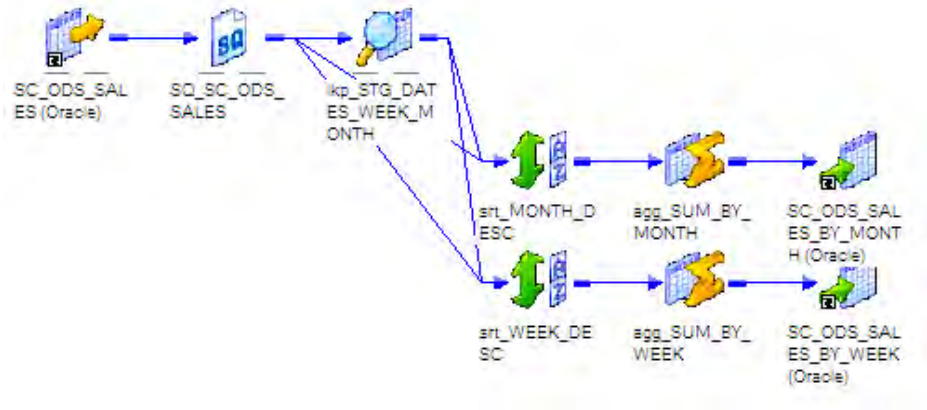
7) Following the same procedure, create a Sorter transformation called **srt_WEEK_DESC** and use it to sort the input to agg_SUM_BY_WEEK.

a) Do not forget to set the Sorted Input box for agg_SUM_BY_WEEK.

b) Alternatively, you can copy and edit the Sorter you have already created

8) The completed Mapping, when you Arrange All Iconic, should look like this:

Figure 80: M9_Sales_Summaries with Sorter Transformations



9) Save your work!

Step 6. Edit the Workflow

1) In Workflow Manager, right-click the session s_m7_Sales_Summaries and select "Refresh Mapping."

Note: "Refresh Mapping" re-reads the mapping information for the Session. If substantial changes have been made to the mapping that might cause it to become invalid, the Workflow Manager marks it invalid just in case.

2) Save the Workflow and verify that it is valid.

3) Run the Workflow.

a) You should get the same Source/Target statistics as before.

b) But compare the run time for the Session to the first run.

Lab 8-1: Using the Debug Wizard

Scenario:

- The Mapping m8_STG_DATES_DEBUG contains at least one error that results in bad data loaded into the target table. You must find and correct this error so that the data warehouse project can proceed successfully.

Goals:

- Use the Debug toolbar
- Use the Debug Wizard

Duration:

35 minutes

Instructions

Step 1. Copy and Inspect the Debug Mapping

1) In the Designer application, make sure you are connected and open to your assigned Dev***xx*** folder.
2) Expand the folder SC_DATA_STRUCTURES
   a) Locate and select the Mapping **m8_STG_DATES_DEBUG**.
   b) From the menu, select **Edit→Copy**.
3) Return to your Dev***xx*** folder.
   a) Select your Mapping subfolder.
   b) From the menu, select **Edit→Paste**.
   c) In the "Copy Confirmation" dialogue, click **Yes**.
4) Save the Repository.

Tip: Note that the Mapping validates properly. The validation process ensures that the Mapping is technically valid, but it cannot test for errors in business logic.

5) Open the Mapping in the Mapping Designer workspace.
   a) Inspect the Mapping to get an overall idea of what kind of processing is being done.
   b) Read the expressions in the Expression transformation.
6) You have been told only that there is an "error" in the data being written to the target, without any further clarification as to the nature of the error.

Tip: Many Mapping errors can be found by carefully inspecting the Mapping, without using the Debug Wizard. If the error cannot be quickly located in this manner, the Debug Wizard can help you by showing the actual data passing through the transformation ports. However, to use the Debug Wizard effectively, you need to understand the logic of the Mapping.

Step 2. Open the Debug Toolbar and Start the Debug Wizard

1) On the Toolbar, right-click to bring up a list of available toolbars. Select the "Debugger" toolbar.

Figure 81: Open the Debugger Toolbar

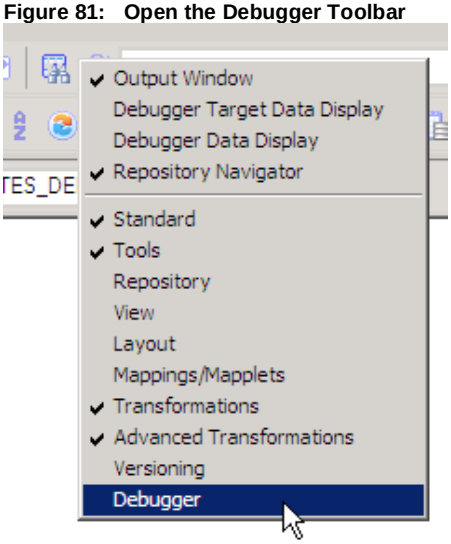
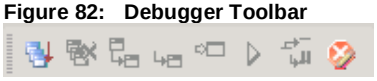


Figure 82: Debugger Toolbar



Tip: If the Debugger Toolbar is not visible, it is possible that another toolbar has shifted it off the screen. Rearrange the other toolbars until you can see it.

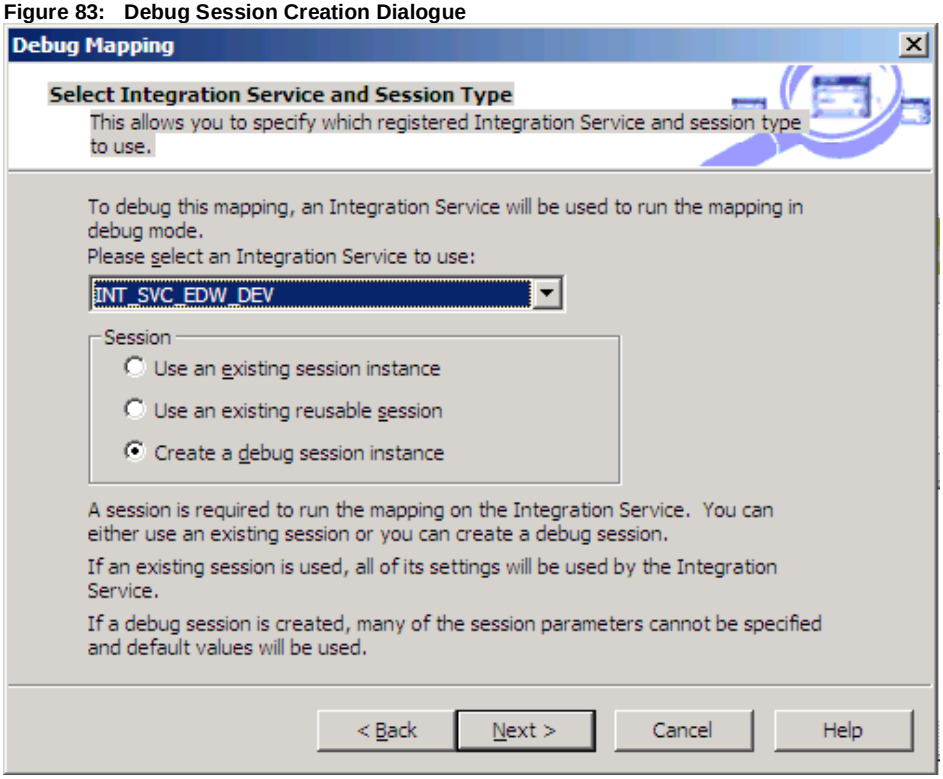
2) EITHER click the Start Debugger button ( ) OR press F9 to start the Debug Wizard.

3) The first page of the Debug Wizard is informational. Please read it and press **Next**.

Tip: The Debug Wizard requires a valid Mapping and Session to run – it cannot help you determine why a Mapping is invalid. The Output window of the Designer will show you the reason(s) why a Mapping is invalid.

4) The Wizard should now look like this:

Figure 83: Debug Session Creation Dialogue



a) From the dropdown box, select the Integration Service INT_SVC_EDW_DEV.

b) In the Session box, select the **Create a debug session** radio button.

c) Click **Next**.

5) The next page of the Wizard allows you to set connection properties, similar to creating Sessions in the Workflow Manager application.

a) Set the Target Connection Value to STG*xx*

(i) You will discard the debugger data in a later step, so this value will be ignored.

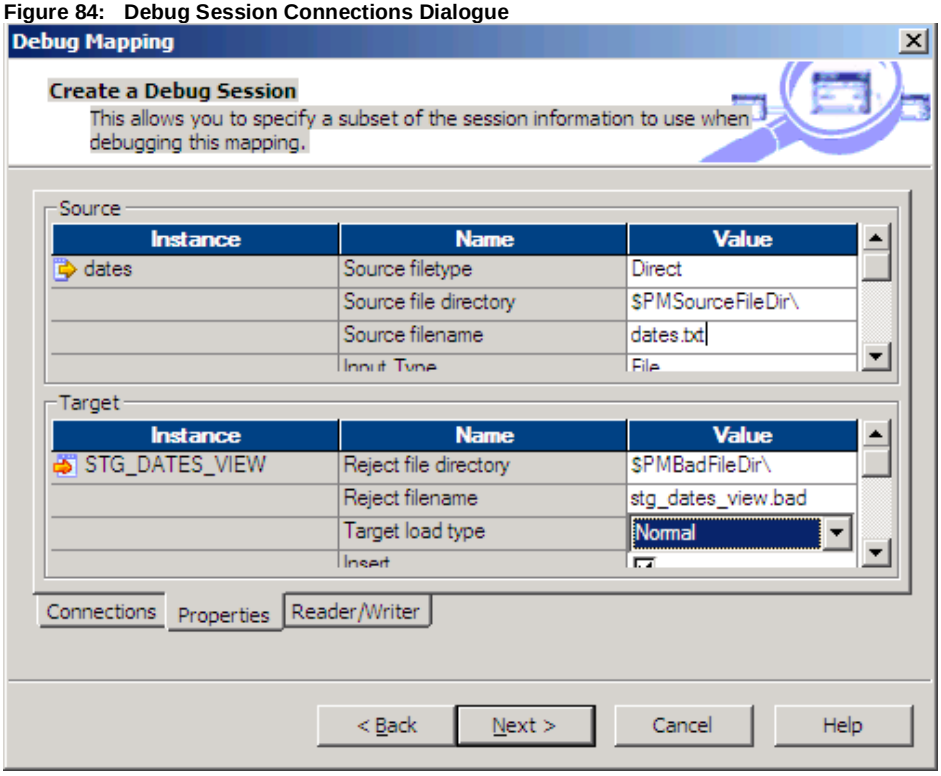
b) Select the Properties tab at the bottom.

(i) Add the ".txt" extension to the Source filename property value.

(ii) Set the Target load type property to Normal

c) Your Wizard should now look like this:

Figure 84: Debug Session Connections Dialogue

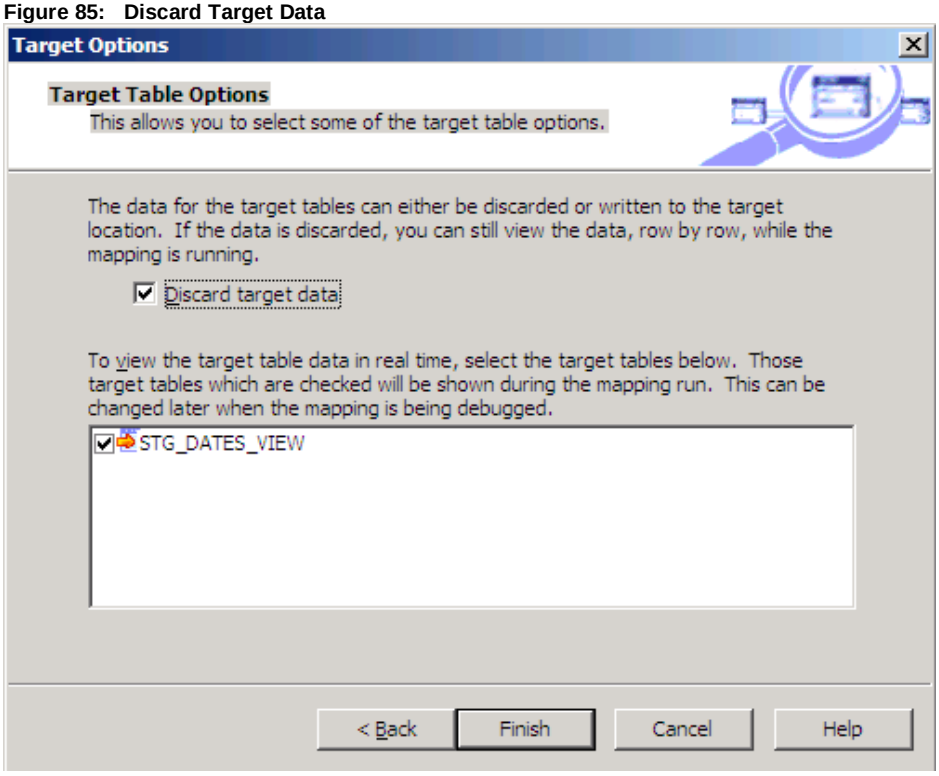


d) Click **Next** three times.

These panels enable you to set which transformations in the Mapping you wish to monitor in this debugging session, and set Session configuration information, such as a parameter file or which connections the variables $Source and $Target correspond to.

e) Ensure that **Discard target data** is checked.

Figure 85: Discard Target Data



f) Click **Finish**.

6) PowerCenter creates and initializes a Debug Session.

a) The Debug Wizard automatically opens windows to view target and transformation data. No data will be read until you are ready to view it.

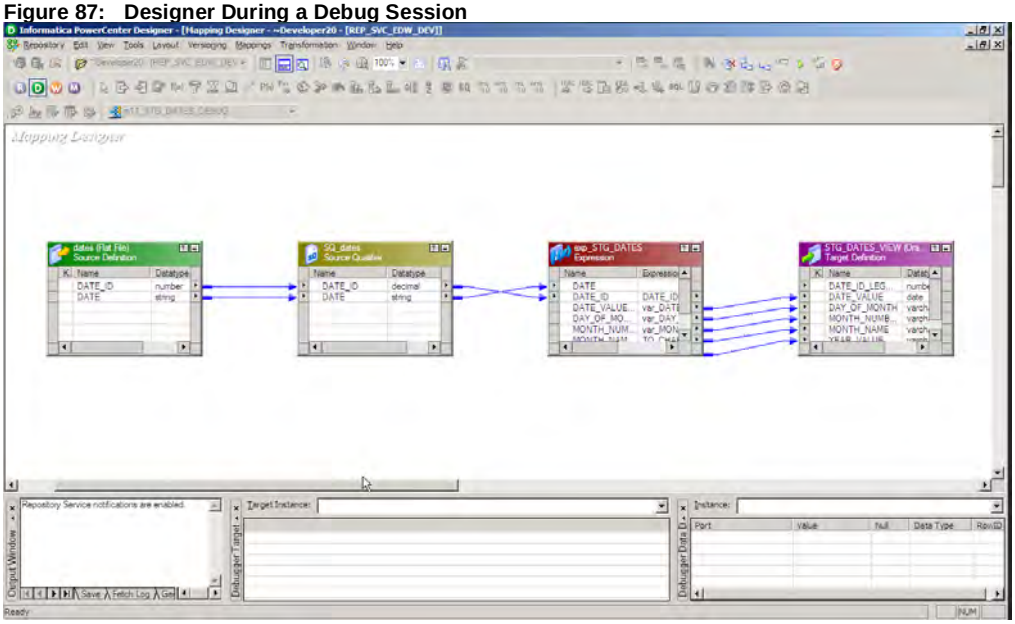
Resize the Debugger Target Data Display and Debugger Data Display windows as needed. A good guideline is to have them look something like this:

Figure 86: Debugger Windows

Step 3. Locate the Error

1) Your screen should now look like this:

Figure 87: Designer During a Debug Session



2) In the Target Instance dropdown box, select **STG_DATES_VIEW**.

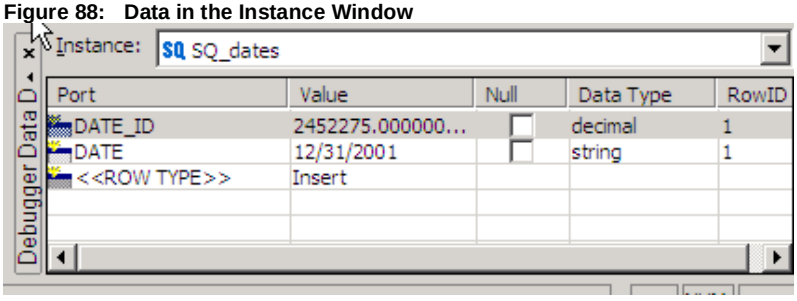
3) In the Instance dropdown box, select **SQ_dates**.

> **Note:** The term "instance" here refers to an object in the Mapping. Thus, each transformation is an "instance."

4) Click the "Next Instance" button ( ) on the Debugger toolbar.

   a) Data displays in the Instance window.

Figure 88: Data in the Instance Window



5) Toggle the dropdown to the Expression transformation.

   a) Note that there is no data available as yet – the Instance window, with the Next Instance button, shows data as it moves from transformation to transformation through the Mapping.

6) Click the Step to Instance button ( )

   a) Note that one more row has been read, and the first row has been "pushed" to the Expression transformation and the Target table.

7) Click the Step to Instance button several more times (at least 13), watching how the data flows from the Instance window to the Target Instance window. Compare the results between the Target instance and Instance windows.

   a) What is the nature of the error in the data being written to the table?

   b) Double-click the Expression transformation to open it.

Tip: Note that the transformation properties are grayed-out. While you can view and copy expressions, you cannot edit the Mapping or its components while the Debugger session is running.

   c) What is causing the error?

Step 4. Fix the Error

Tip: Nonetheless, you CAN try new variations on expression while the Debugger is running.

1) Click the Ports tab.

   a) Enter the Expression Editor for one of the output ports – preferably the one that seems likely to be causing the problem.

   b) Select the text of the expression (even though it is grayed-out) and copy it to the Windows clipboard by typing Ctrl+C.

   c) Click **Cancel** twice.

   d) Right-click the Expression transformation and select **Evaluate Expression**.

   e) Paste the expression text you chose into the Expression Editor and press **Evaluate**.

      (i) The Debugger will immediately evaluate the expression with the current data in the ports.

      (ii) You can make as many changes to the Expression here as you need.

      (iii) Once you have a modified expression that you want to keep, copy it to the Windows clipboard.

2) Close the expression evaluator

3) Stop the Debugger by pressing the Stop Debugger button () on the Debugger toolbar

   a) Click **Yes** to "Shutdown the debugger."

4) Edit the Expression transformation and put your modified Expression in place by pasting it into the Expression Editor.

5) Save your work.

6) Restart the Debugger and test to ensure that your fix worked.

Answers

3.7.a What is the nature of the error in the data being written to the table?
The month and date seem to be reversed. That is, the data comes in as January 1, January 2, etc., but is being written as January 1, February 1, etc.

3.7.c. What is causing the error?
The Expression Editor is using a format of DD/MM/YYYY but the incoming data has a format of MM/DD/YYYY.

Lab 9-1: Updating Target Tables

Scenario:

- The ODS_Customer table needs to be updated periodically with a list of customers that have recently done business with Mersche Motors.
- Many customers will be duplicates of existing customers. Some will be the same people but with new information.
- You need to devise a Mapping that will identify the new customers and add them to the ODS table.
- Also, if a customer's information has been updated, the ODS table needs to be updated as well.

Goals:

- Use a single Source definition to read two files and combine their data in a single data stream.
- Remove duplicate rows.
- Create logic that
  - Rejects the record if the incoming CUSTOMER_ID is missing
  - Inserts the record if the customer does not already exist in ODS_CUSTOMERS
  - Updates the record if the customer already exists in ODS_CUSTOMERS

Duration:

90 minutes

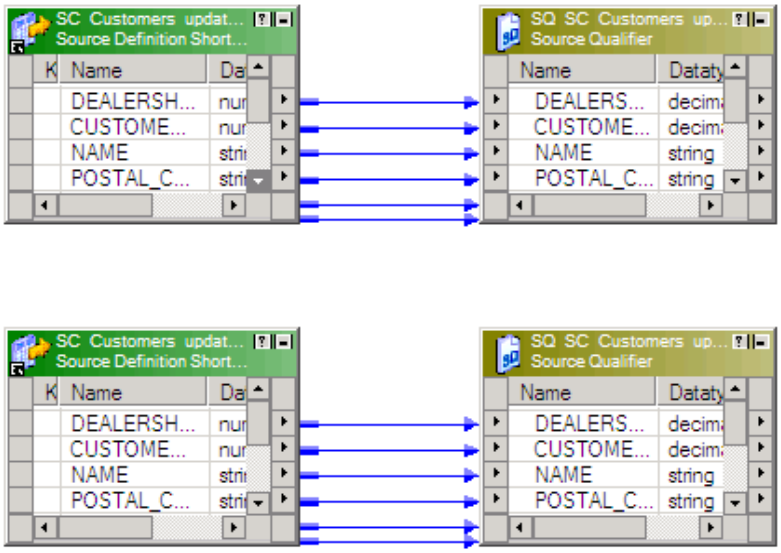
Instructions

Step 1. Sources and Targets

1) Create a new Mapping named **m9_Update_Customers_*xx***.
2) Drag the Source definition **SC_Customers_updated** into the mapping.
   a) Rename the Source definition as **SC_Customers_updated_East**
   b) Rename the Source Qualifier as **SQ_SC_Customers_updated_East**
3) Drag the Source definition **SC_Customers_updated** into the mapping again
   a) Rename the Source definition as **SC_Customers_updated_West**
   b) Rename the Source Qualifier as **SQ_SC_Customers_updated_West**
4) Arrange All

5) The Mapping should look like this:

Figure 89: Two Sources Based on One Source Definition



Step 2. Merge the Two Data Streams Using a Union Transformation

1) Use the Union icon ( ) to add a Union transformation to the Mapping.

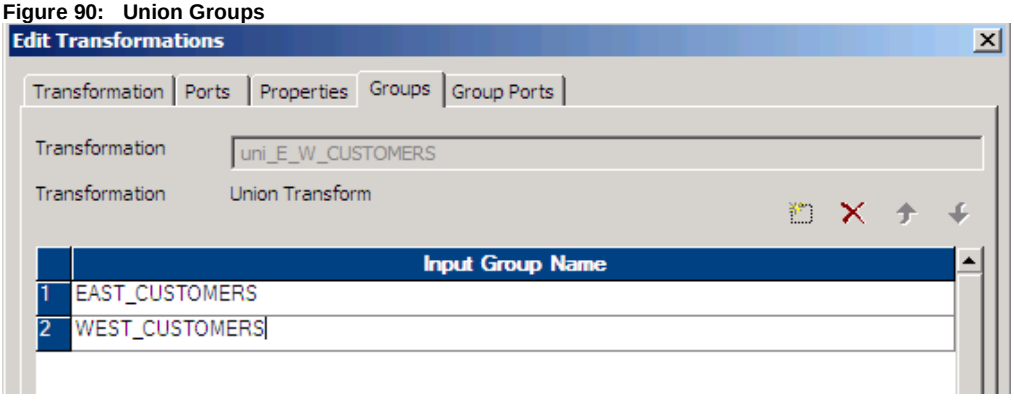
2) Edit the Union transformation.

   a) Rename it uni_E_W_CUSTOMERS.

   b) Select the Groups tab.

      (i) Configure two groups named **EAST_CUSTOMERS** and **WEST_CUSTOMERS**.

Figure 90: Union Groups

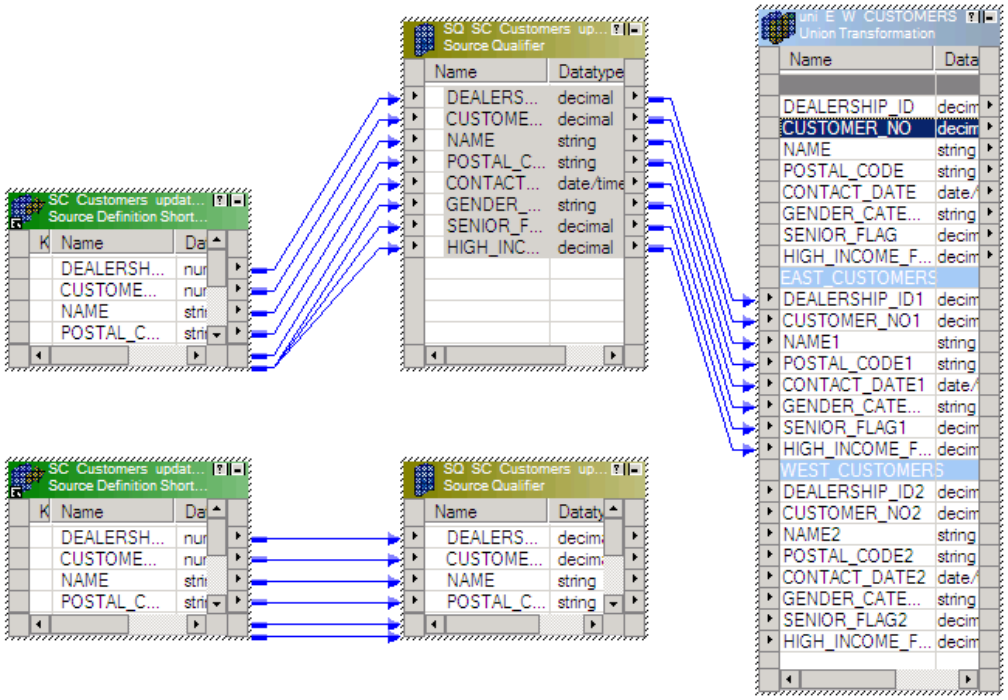


   c) Click **OK**.

3) Select all the ports in SQ_SC_Customers_updated_East and drag them to the Union transformation.

a) The Mapping should now look like this:

Figure 91: Mapping with Union



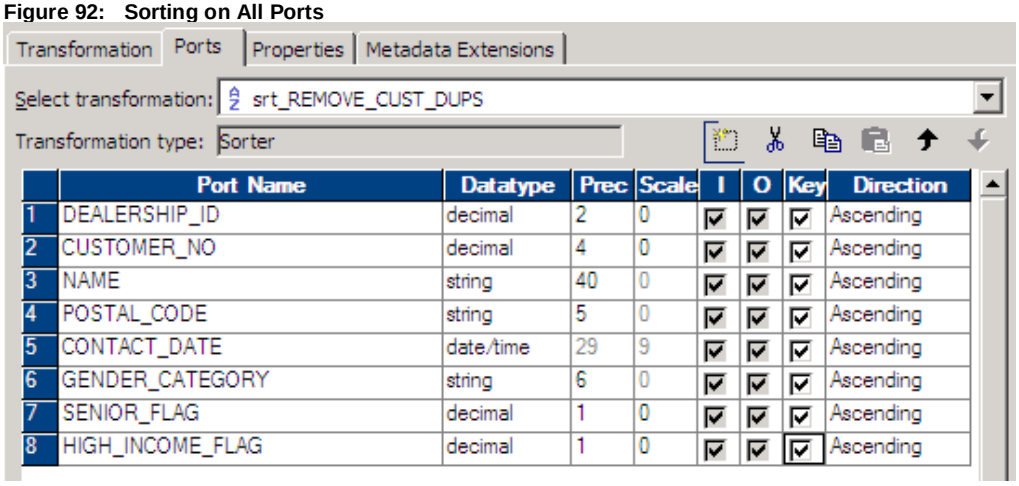
4) Link the ports from SQ_SC_Customers_updated_West to the WEST_CUSTOMERS group of the Union transformation.

Step 3. Remove Duplicates with a Sorter Transformation

1) Add a Sorter transformation to the Mapping.
2) Drag all the output ports from the Union transformation to it.
3) Edit the Sorter transformation.
   a) Rename it **srt_REMOVE_CUST_DUPS**.

b) Sort on *all* the ports.

Figure 92: Sorting on All Ports

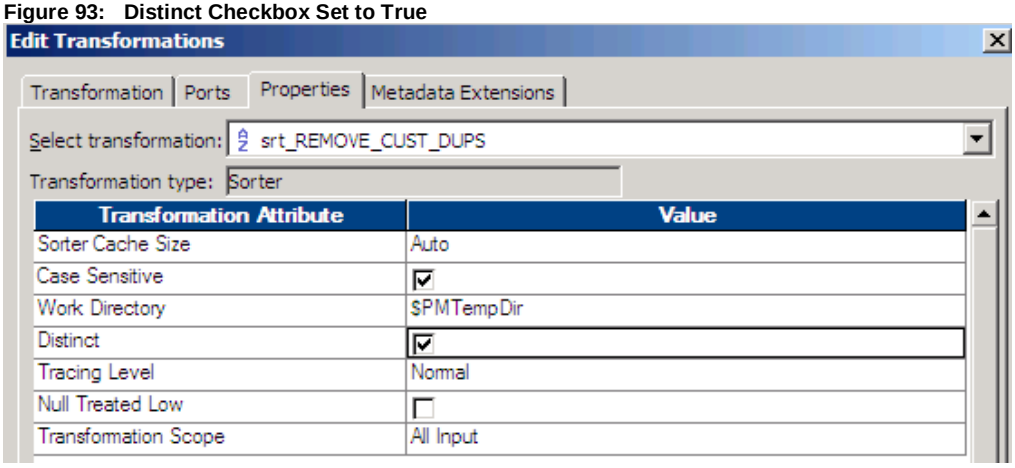


	Port Name	Datatype	Prec	Scale	I	O	Key	Direction
1	DEALERSHIP_ID	decimal	2	0	☑	☑	☑	Ascending
2	CUSTOMER_NO	decimal	4	0	☑	☑	☑	Ascending
3	NAME	string	40	0	☑	☑	☑	Ascending
4	POSTAL_CODE	string	5	0	☑	☑	☑	Ascending
5	CONTACT_DATE	date/time	29	9	☑	☑	☑	Ascending
6	GENDER_CATEGORY	string	6	0	☑	☑	☑	Ascending
7	SENIOR_FLAG	decimal	1	0	☑	☑	☑	Ascending
8	HIGH_INCOME_FLAG	decimal	1	0	☑	☑	☑	Ascending

c) Select the Properties tab.

d) Set the Distinct checkbox to true (checked)

Figure 93: Distinct Checkbox Set to True

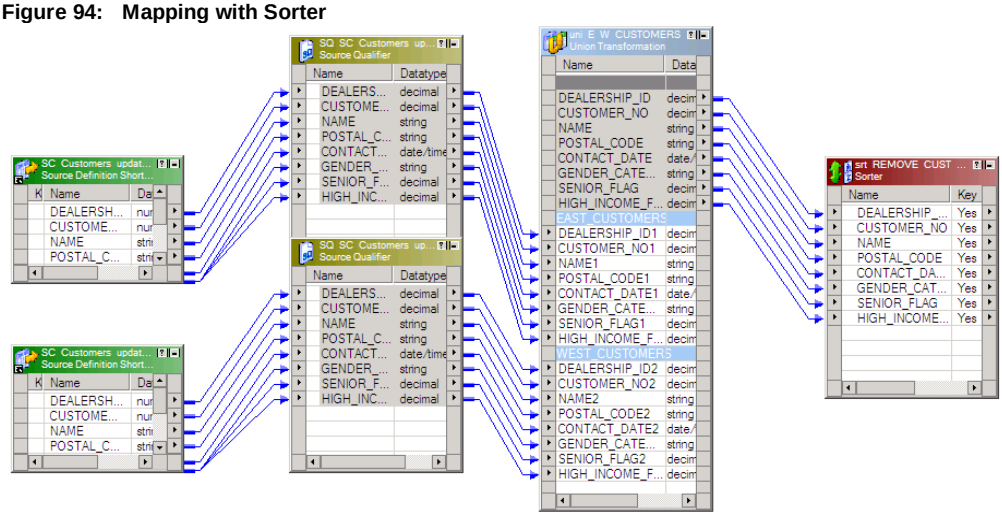


Transformation Attribute	Value
Sorter Cache Size	Auto
Case Sensitive	☑
Work Directory	$PMTempDir
Distinct	☑
Tracing Level	Normal
Null Treated Low	☐
Transformation Scope	All Input

e) Click **OK**.

f) The Mapping should look like this:

Figure 94: Mapping with Sorter



4) Iconize the Source definitions, Source Qualifiers, and Union transformation, and arrange the Mapping to give you space on the right.

Step 4. Create and Configure a Lookup on the ODS_CUSTOMERS target table

1) Drag the Target **SC_ODS_CUSTOMERS** into the Mapping.

2) Add a Lookup transformation to the Mapping.

   a) Use the relational table SC_ODS_CUSTOMERS.

3) Drag the port CUSTOMER_NO from the Sorter transformation to the Lookup transformation.

Note: the rule of "active vs. passive" transformation objects applies here. The Sorter is an active transformation. Therefore, it cannot be bypassed by bringing this port directly through from the Union transformation to the Target.

4) Edit the Lookup transformation.

   a) Rename it **lkp_ODS_CUSTOMERS**.

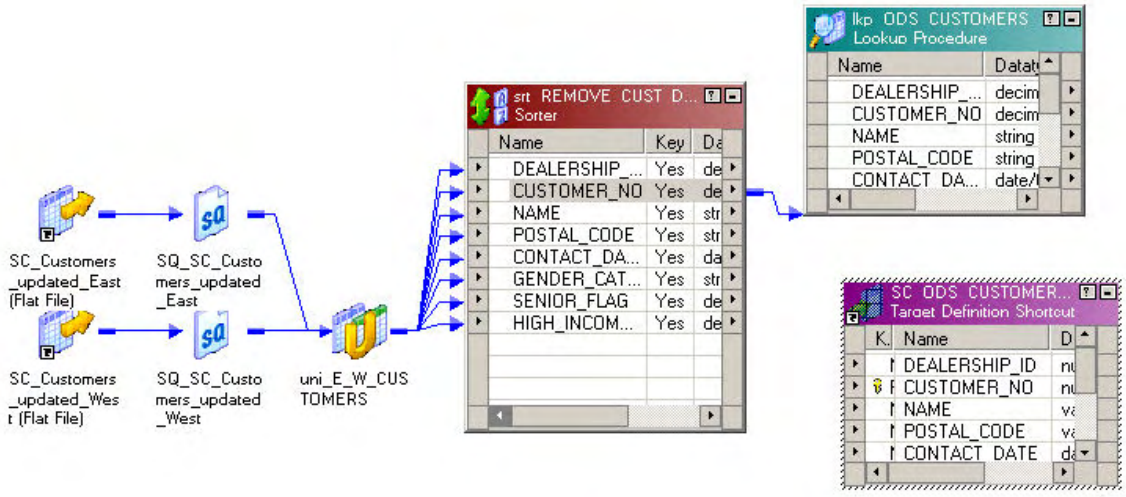
   b) Rename the port CUSTOMER_NO1 to **CUSTOMER_NO_SOURCE**

   c) Set the Lookup condition to **CUSTOMER_NO = CUSTOMER_NO_SOURCE**

   d) Click **OK**.

5) The Mapping should now look like this:

Figure 95: Mapping with Lookup



Step 5. Create and Configure an Update Strategy Transformation

1) Use the Update Strategy icon ( ) to add an Update transformation to the Mapping.

2) Drag all the ports from the Sorter transformation to the Update Strategy transformation.

3) Drag the CUSTOMER_NO port from the Lookup transformation to the Update Strategy transformation.

4) Edit the Update Strategy transformation.

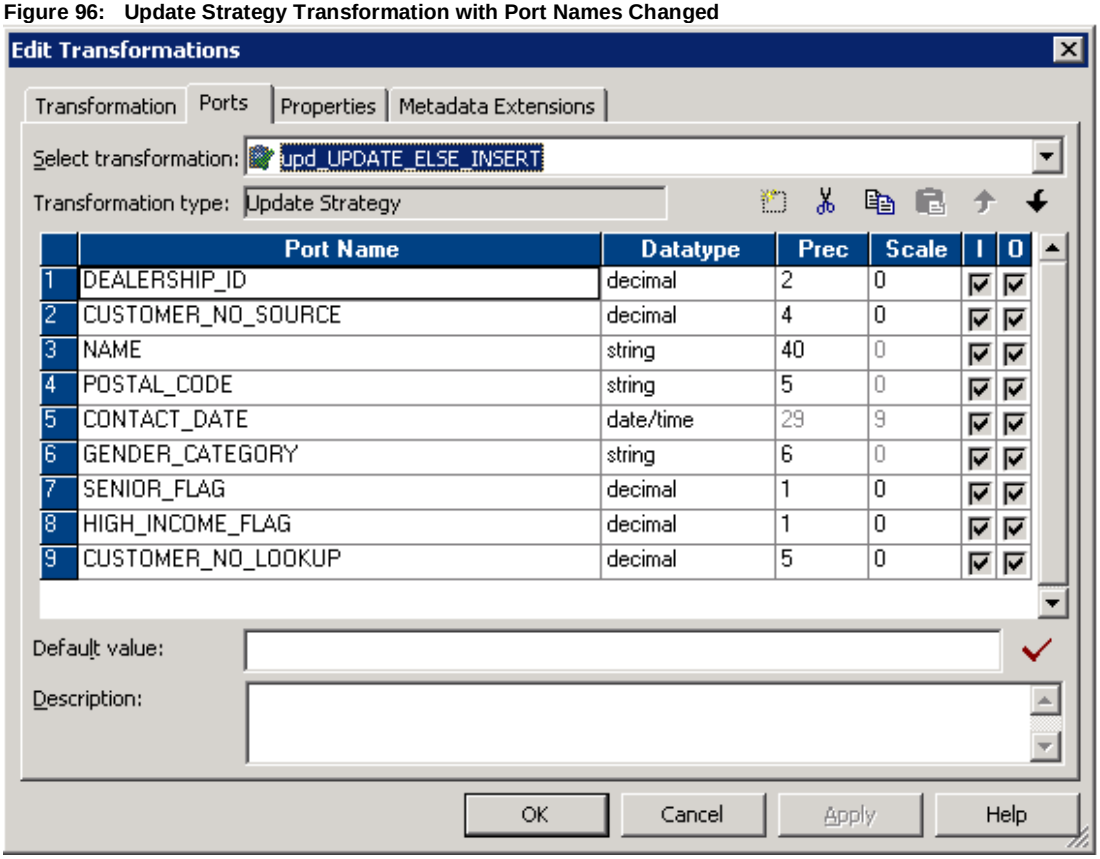
   a) Rename it to **upd_UPDATE_ELSE_INSERT**.

   b) Select the Ports tab.

   c) Change the name of the CUSTOMER_NO port (the one coming from the Sorter) to **CUSTOMER_NO_SOURCE**

   d) Change the name of the CUSTOMER_NO1 port (the one coming from the Lookup) to **CUSTOMER_NO_LOOKUP**

Figure 96: Update Strategy Transformation with Port Names Changed



e) Select the Properties tab.

f) In the Update Strategy Expression field, enter the Expression Editor.

Update Strategy Expression

The goal is to

Reject the row if the customer number is null
Insert the row if the customer number is new
Update the record if the customer number already exists.

Starting with the case where the Customer number coming from the Source is null, we want to reject the record:

IIF(ISNULL(CUSTOMER_NO_SOURCE), DD_REJECT

Then we proceed to the case where the customer number from the source exists and does not exist in ODS_CUSTOMER, meaning it is a new customer. Then we want to insert a new record:

IIF(ISNULL(CUSTOMER_NO_LOOKUP), DD_INSERT

If neither of these is true, then the customer number already exists in ODS_CUSTOMER, so we want to update it:

DD_UPDATE

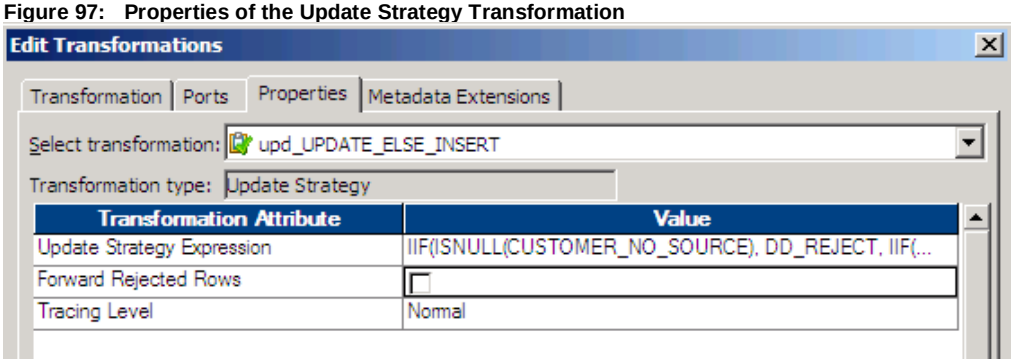
When we put it all together, the expression becomes:

IIF(ISNULL(CUSTOMER_NO_SOURCE), DD_REJECT,
IIF(ISNULL(CUSTOMER_NO_LOOKUP), DD_INSERT, DD_UPDATE))

g) Enter the above expression in the Expression Editor and click **Validate**.

(i) If the expression does not validate, fix it.

Figure 97: Properties of the Update Strategy Transformation



h) Click **OK** to exit the Edit Transformations dialogue.

5) Use Autolink by Position to connect the ports from the Update Strategy transformation to the Target definition.

6) Arrange All Iconic to view the logic of the Mapping.

Figure 98: M9_Update_Customers_*xx*

Step 6. Create and Run the Workflow

1) Create a Workflow named **wkf_Update_ODS_CUSTOMERS_*xx***
2) The source files will be customers_updated_east.dat and customers_updated_west.dat
3) The Target and Lookup transformation will use the Relational connection ODS***xx***
4) Make sure to set Target load type to Normal.
5) ***DO NOT*** set the Truncate target table option. If you do, the existing data in the table will be deleted and the update logic will not work properly.

Figure 99: Source/Target Statistics for wkf_Update_ODS_Customers_*xx*

s_m9_Update_Customers_20 [11/18/2008 12:29:12 PM]

Task Details

Attribute Name	Attribute Value
Instance Name	s_m9_Update_Customers_20
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	NODE01_INTDEVTEST
Start Time	11/18/2008 12:29:12 PM

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_CUST...	NODE01_I...	6106	6295	22	1574	162122
SQ_SC_Custome...	NODE01_I...	6160	6160	0	6160	628320
SQ_SC_Custome...	NODE01_I...	6054	6054	0	6054	617508

Note 1: The specific number of rows may vary depending on whether you did the Extra Credit exercise in Lab 3-1.

Note 2: The number of rejected rows shown here does not reflect the number of rows rejected by the Mapping. Rather, it shows that no errors were thrown by the database. This is to be expected because the Mapping did not forward any rows with a null key field to the database. To see the number of rows actually rejected by the Mapping, you must consult the Session Log.

Lab 10-1: Filling in Missing Data

Scenario:

- IT and test users have noted that there are invalid and missing Postal codes for some customers. This hampers Marketing's efforts to identify who Mersche Motors customers are and how to reach out to them, and must be corrected.
- Marketing wishes customers to be categorized based on a combination of income level and location.
- Marketing wants a file of customers with subprime credit ratings that can be analyzed further.

Goals:

- Use an Unconnected Lookup transformation to attempt to find in which city customers purchased their cars.
  - This works on the assumption that customers purchase their cars locally.
- Use a Router transformation to categorize customers on income level, Postal code, and city into High Income, Sub-Prime, and Uncategorized categories.
  - Since the reality is that Sub-Prime and Uncategorized will be treated the same from a lending standpoint, write them to the same table.
- Override the Sub-Prime/Uncategorized relational writer to write out to a flat file (.CSV) so that the data can easily be loaded into a spreadsheet for further analysis.

Duration:

60 minutes

Instructions

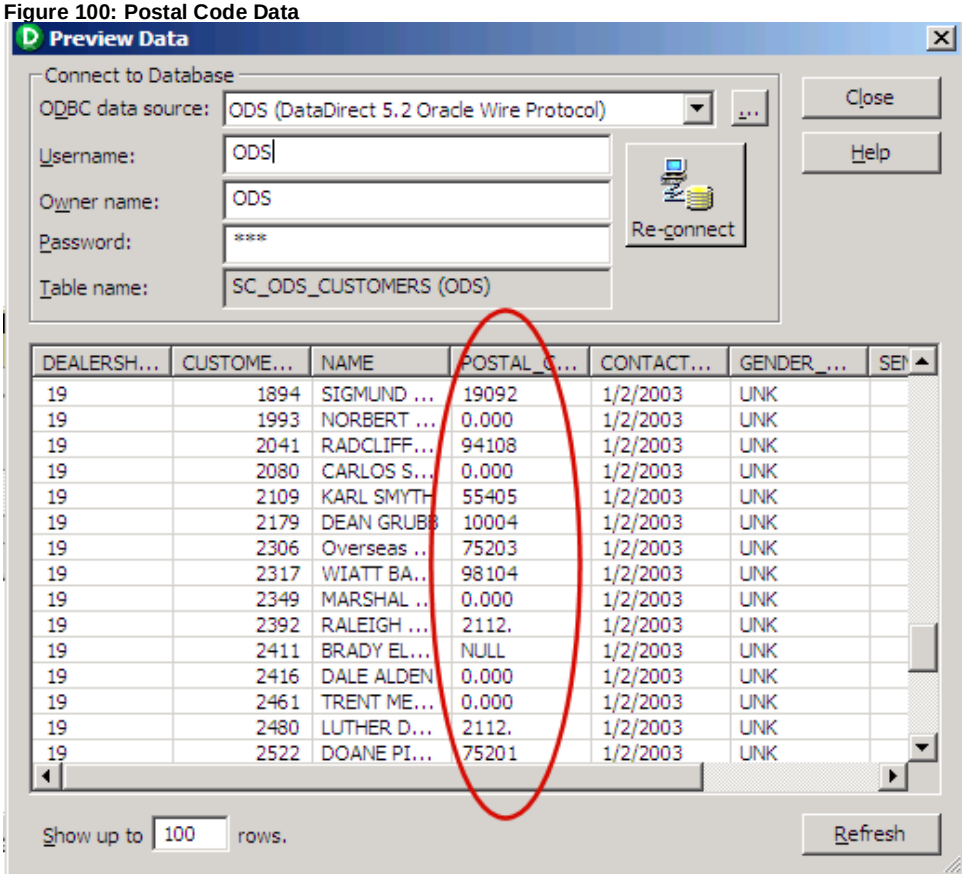
Step 1. Create a Mapping, Select Sources and Targets

1) Create a Mapping called **m10_Customer_Category_*xx***.
2) Add the Source definition **SC_ODS_CUSTOMERS**.
3) Add the Target definitions **SC_ODS_CUSTOMERS_HIGH_VALUE** and **SC_ODS_CUSTOMERS_SUBPRIME**.
4) Save the Mapping (it will not be valid).

Step 2. Identify Bad Data and Establish a Method of Remediation

1) Preview the data in ODS_CUSTOMERS.

Figure 100: Postal Code Data



2) Note that there are bad entries in the Postal Code field, such as:
   0.000
   2112.
   NULL (string)

3) Not every record will require an attempt to repair bad Postal Code.
   a) Therefore, we will use an Unconnected Lookup as needed to fix the data.

4) Close the Preview Data window.

Step 3. Create and Configure a Lookup Transformation

1) Create a Lookup transformation named **lkp_RETURN_LOCATION**.

2) Base the Lookup on the SC_STG_DEALERSHIP table.

3) Drag DEALERSHIP_ID from the Source Qualifier to the Lookup.

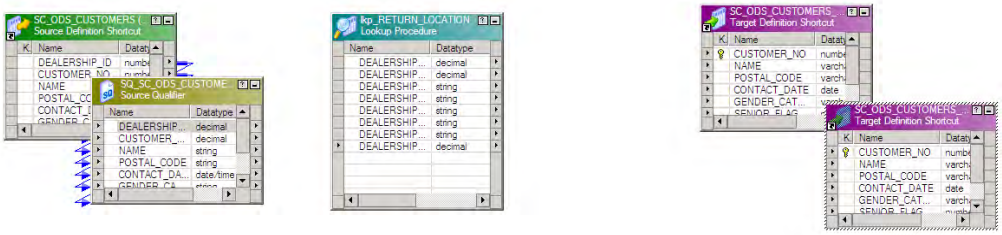
4) Edit the Lookup transformation.
   a) Rename the Port from DEALERSHIP_ID1 to DEALERSHIP_ID_SOURCE
   b) Set the Lookup condition as
      DEALERSHIP_ID = DEALERSHIP_ID_SOURCE
   c) Set DEALERSHIP_LOCATION as the Return port.

d) Click **OK**.

5) Delete the link from the Source Qualifier to the Lookup transformation.

6) The Mapping should now look like this:

Figure 101: Mapping with Unconnected Lookup Added



Step 4. Create and Configure an Expression Transformation

1) Create an Expression transformation.

2) Drag all the ports from the Source Qualifier into the Expression transformation.

3) Edit the Expression transformation.

   a) Rename it **exp_FIND_LOCATION**.

   b) Create a new Output port named LOCATION with datatype STRING and precision 20.

   c) Enter the Expression Editor for the port LOCATION.

      (i) In the Functions tab, scroll to the bottom of the list of functions.

      (ii) At the bottom of the list you will see a folder named "Lookups." Open this folder and you will see the unconnected Lookup transformation you just created.

      (iii) Create the expression so that if the value of POSTAL_CODE is NULL, or contains the string NULL, or is equal to 0.000, then look up the POSTAL_CODE in the STG_DEALERSHIP table based on DEALERSHIP_ID. Otherwise, return the value of POSTAL_CODE.

      ```
      IIF(((POSTAL_CODE !=  '0.000') AND NOT
      ISNULL(POSTAL_CODE) AND (POSTAL_CODE != 'NULL')),
      POSTAL_CODE,
      :LKP.LKP_RETURN_LOCATION(DEALERSHIP_ID) )
      ```

      (iv) Make sure your expression is valid, then click **OK**.

   d) Click **OK** to exit the Edit Transformations dialogue.

Step 5. Create and Configure a Router Transformation to Classify Customers

1) Create a Router transformation ()

2) Drag all ports from the Expression transformation to the Router transformation ***except*** DEALERSHIP_ID.

3) Edit the Router transformation.

   a) Rename it rtr_**CLASSIFY_CUSTOMERS**.

   b) Create two Groups, named **HIGH_VALUE** and **SUBPRIME**.

   c) For the HIGH_VALUE group set the filter condition to send records to this group when
   the High Income flag is set to 1
   OR
   the value in the Location port is 19104, 10005, 90004, Newport Beach, Scottsdale, or West Palm Beach

   d) For the SUBPRIME group set the condition to send records to this group when the High Income flag is set to 0
   AND
   the value in the Location port is 55409, 98112, 75201, Indianapolis, or Phoenix.

> **Note:** Obviously these location choices are very simplistic. They are acceptable for illustrating the use of the Router transformation in this exercise, however.

   e) Verify that your expressions are valid.

Step 6. Connect Router Output Groups to Targets

1) Connect the appropriate HIGH_VALUE output group ports to the target SC_ODS_CUSTOMERS_HIGH_VALUE

2) Connect the SUBPRIME group to the target SC_ODS_CUSTOMERS_SUBPRIME

3) To connect the Default group (uncategorized records)

   a) Drag in another copy of SC_ODS_CUSTOMERS_SUBPRIME

   b) Rename it **SC_ODS_CUSTOMERS_UNCATEGORIZED**

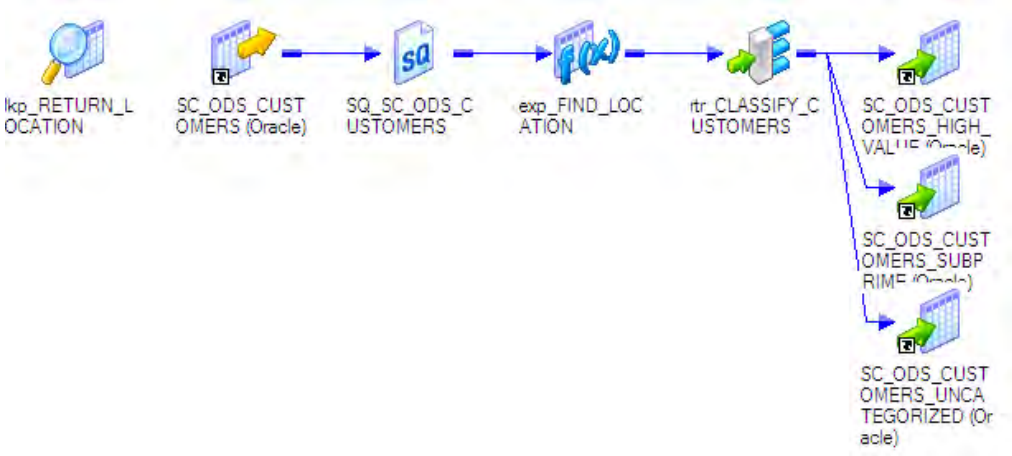
> **Note:** Even though the Target instance has been renamed, it will still write to the original table name. You can verify this by looking at the Shortcut To fields.

   c) Connect the DEFAULT group from the Router to the Target SC_ODS_CUSTOMERS_UNCATEGORIZED

4) Save your work.

5) Arrange All Iconic.

Figure 102: M10_Customer_Category_*xx*



Step 7. Create and Run a Workflow for the Mapping

1) Name the Workflow wkf_Customer_Category_***xx***.

Note: You will override the Relational Writer so that the Subprime and Uncategorized customers are written to .csv files.

2) Edit the Session object.

   a) Select the Mapping tab.

   b) Select the Target SC_ODS_CUSTOMERS_SUBPRIME

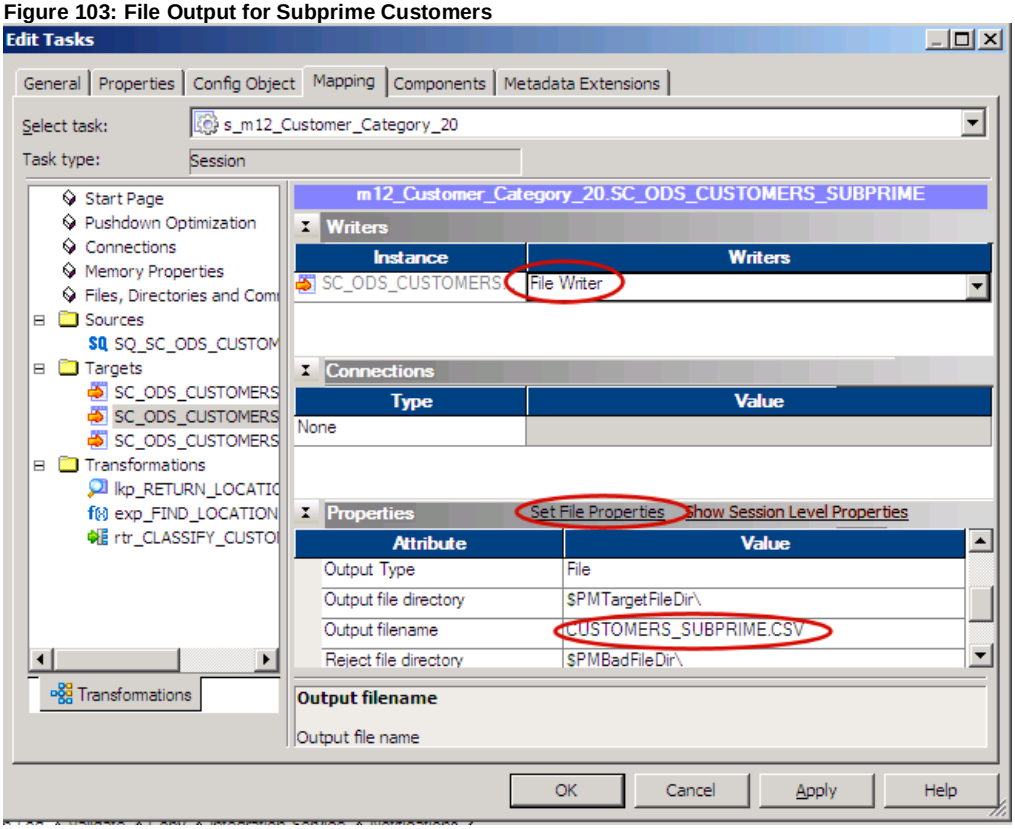
   c) Using the drop box, set the value of the Writers property to **File Writer**.

   d) In the Properties window, scroll down to find the Output Filename attribute.

   e) Change its value to **CUSTOMERS_SUBPRIME_*xx*.csv**

f) The Edit Tasks window should look like this:

Figure 103: File Output for Subprime Customers



3) In the "Properties" bar, click the link **Set File Properties**

   a) In the "Flat Files – Targets" dialogue, select **Delimited** and click **Advanced**.

   b) In the "Delimited File Properties – Targets" dialogue, make sure that the Column Delimiter is a comma (,) character.

   c) Click **OK**

   d) Click **OK**.

4) Using the same procedure, set the Target SC_ODS_CUSTOMERS_UNCATEGORIZED to output to a file named **CUSTOMERS_UNCATEGORIZED_*xx*.csv**

5) Save and run the Workflow.

Figure 104: Source/Target Statistics

Source/Target Statistics

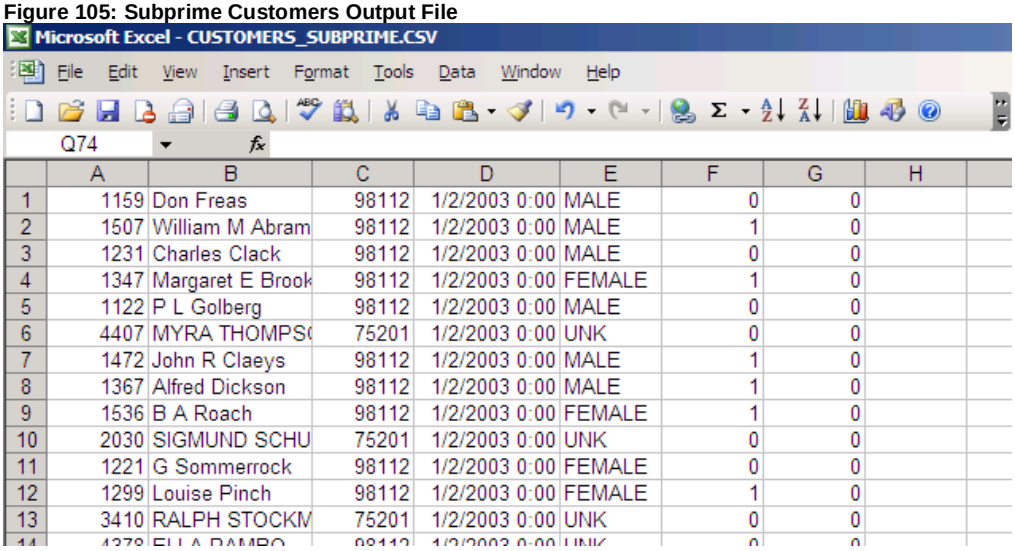
Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_CUST...	node01_...	4724	4724	0	4724	510192
SC_ODS_CUST...	node01_...	26	26	0	26	2808
SC_ODS_CUST...	node01_...	1373	1373	0	1373	148284
SQ_SC_ODS_C...	node01_...	6123	6123	0	6123	630669

6) In Windows Explorer, navigate to C:\pmfiles\TgtFiles (or the equivalent using your mapped drive, InfaShared on IntDevTest).

7) Double-click the file **CUSTOMERS_SUBPRIME_*xx*.csv** to open it in MS Excel. (If you don't have MS Excel, then use an editor of your choice such as Notepad.)

   a) It should look like this:

Figure 105: Subprime Customers Output File



	A	B	C	D	E	F	G	H
1	1159	Don Freas	98112	1/2/2003 0:00	MALE	0	0	
2	1507	William M Abram	98112	1/2/2003 0:00	MALE	1	0	
3	1231	Charles Clack	98112	1/2/2003 0:00	MALE	0	0	
4	1347	Margaret E Brook	98112	1/2/2003 0:00	FEMALE	1	0	
5	1122	P L Golberg	98112	1/2/2003 0:00	MALE	0	0	
6	4407	MYRA THOMPS(	75201	1/2/2003 0:00	UNK	0	0	
7	1472	John R Claeys	98112	1/2/2003 0:00	MALE	1	0	
8	1367	Alfred Dickson	98112	1/2/2003 0:00	MALE	1	0	
9	1536	B A Roach	98112	1/2/2003 0:00	FEMALE	1	0	
10	2030	SIGMUND SCHU	75201	1/2/2003 0:00	UNK	0	0	
11	1221	G Sommerrock	98112	1/2/2003 0:00	FEMALE	0	0	
12	1299	Louise Pinch	98112	1/2/2003 0:00	FEMALE	1	0	
13	3410	RALPH STOCKM	75201	1/2/2003 0:00	UNK	0	0	

Extra Credit

1. Extend the invalid POSTAL_CODE search to include fields that have a period character (.)
2. Redesign the mapping so that all of the POSTAL_CODE values are replaced with city names.

Lab 10-2: Using a Persistent Variable

Scenario:

- New employees need to be added to ODS_EMPLOYEES
- They must have unique IDs assigned to them

Goals:

- Further leverage the reusable transformation EXP_Format_Records
- Use a persisted Repository variable to hold and increment the latest employee identification number

Duration:

45 minutes

Instructions

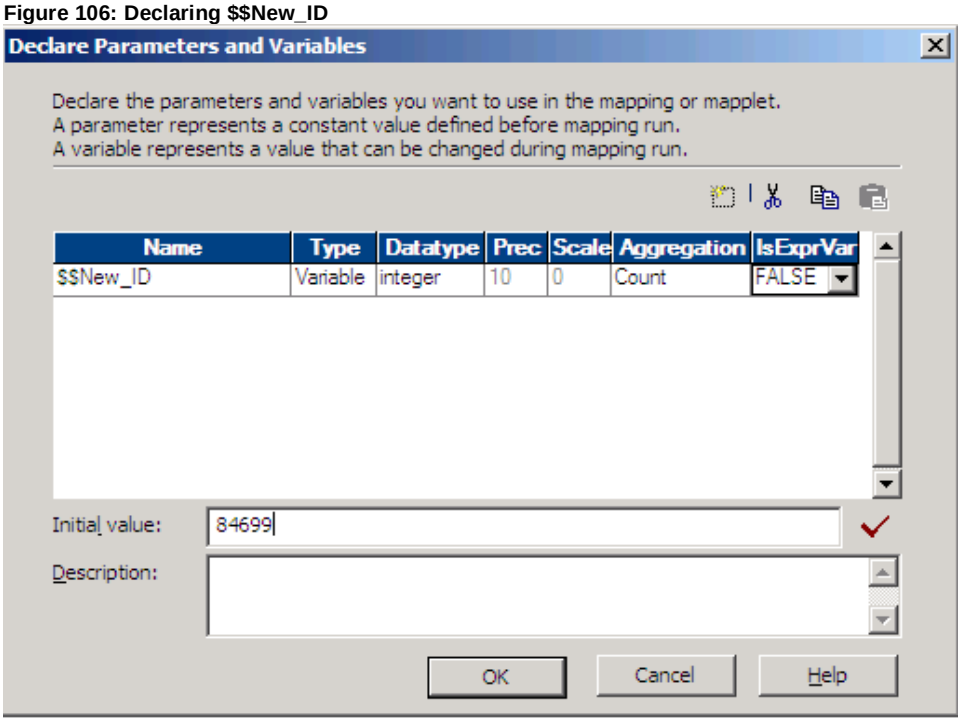
Step 1. Create a Mapping Variable

1) Create a new Mapping named **m10_Load_New_Employees_Summary_*xx***.
2) Drag in the flat file Source definition **SC_Employees**.
3) Drag in the relational Target definition **SC_ODS_NEW_EMPLOYEE_SUMMARY**.
4) From the menu select **Mappings → Parameters and Variables**.
   a) Create a new Variable with the following properties:

Name:	$$New_ID
Type:	Variable
Datatype:	Integer
Aggregation Type:	Count
IsExprVar:	FALSE
Initial value:	84699

Note: This variable will be incremented and used to generate new employee IDs.

Figure 106: Declaring $$New_ID



Step 2. Add and Edit the Expression Transformation

In this step, you will make a non-reusable copy of the reusable transformation you created earlier in the course, and edit it.

1) In your folder, expand the Transformations subfolder.
2) Make a non-reusable copy of a reusable transformation
   a) Select the re_exp_Format_Persons transformation
   b) Click and hold the click, then hold down the CTRL key.
   c) Now drag the transformation into the open mapping, m13_Load_New_Employees_Summary_*xx*.
   d) Notice that while you are doing this the Status Bar reads, "Make a non-reusable copy of this transformation and add it to this mapping."
   e) This is how you make a non-reusable copy of a reusable transformation.
3) Edit the Expression transformation.
   a) Change its name to exp_Format_New_Employees
   b) Delete the ports Income and High Income Flag.
   c) Add a new output port named NEW_EMPLOYEE_ID, datatype Integer.
4) Edit the expression for the new port.
   a) Open the expression editor and delete the existing expression (NEW_EMPLOYEE_ID).
   b) Select the Functions tab.
      (i) Scroll to the bottom of the Functions list.

(ii) Open the folder named Variables.

(iii) Double-click the function **SetCountVariable** to place it in the Expression Editor.

c) Select the Variables tab.

(i) Open the folder named Mapping Variables.

(ii) Double-click the variable **$$New_ID** to add it to the expression.

(iii) The Expression should now read:
SETCOUNTVARIABLE($$New_ID)

(iv) Click **OK** to validate the expression and exit the Expression Editor.

d) Click **OK** to exit the Edit Transformations dialogue.

5) Link the following ports from the Source Qualifier to the Expression transformation:

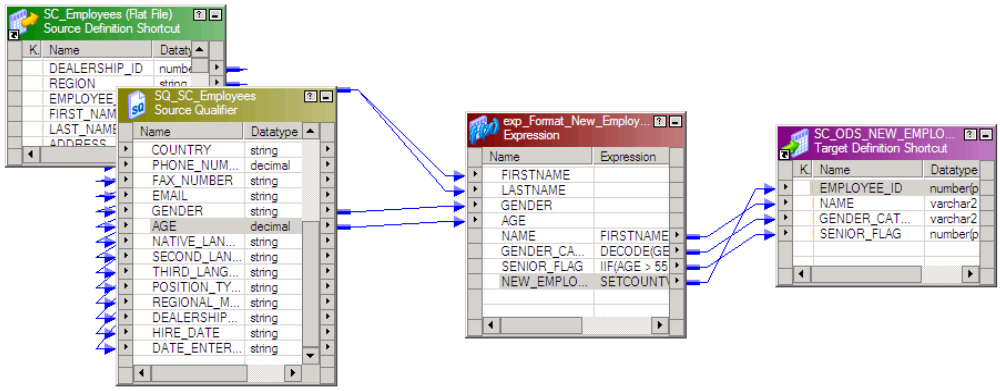
From Port	To Port
FIRST_NAME	FIRSTNAME
LAST_NAME	LASTNAME
GENDER	GENDER
AGE	AGE

6) Link the following ports from the Expression transformation to the Target:

From Port	To Port
NAME	NAME
GENDER_CATEGORY	GENDER_CATEGORY
SENIOR_FLAG	SENIOR_FLAG
NEW_EMPLOYEE_ID	EMPLOYEE_ID

7) The Mapping should now look like this:

Figure 107: m10_Load_New_Employees_Summary_*xx*



Step 3. Create a Workflow and Run the Mapping

a) Name the Workflow **wkf_Load_New_Employees_Summary_*xx***.

b) The source file is named **employees_new.dat**.

c) The target relational connection is ODS***xx***.

d) DO NOT set the Truncate option.

Figure 108: Source/Target Statistics for s_m10_Load_New_Employees_Summary

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows	Throughput (Rows/Sec)	Throughput (Bytes/Sec)
SC_ODS_NEW_...	NODE01_I...	109	109	0	109	7085
SQ_SC_Employees	NODE01_I...	109	109	0	109	3706

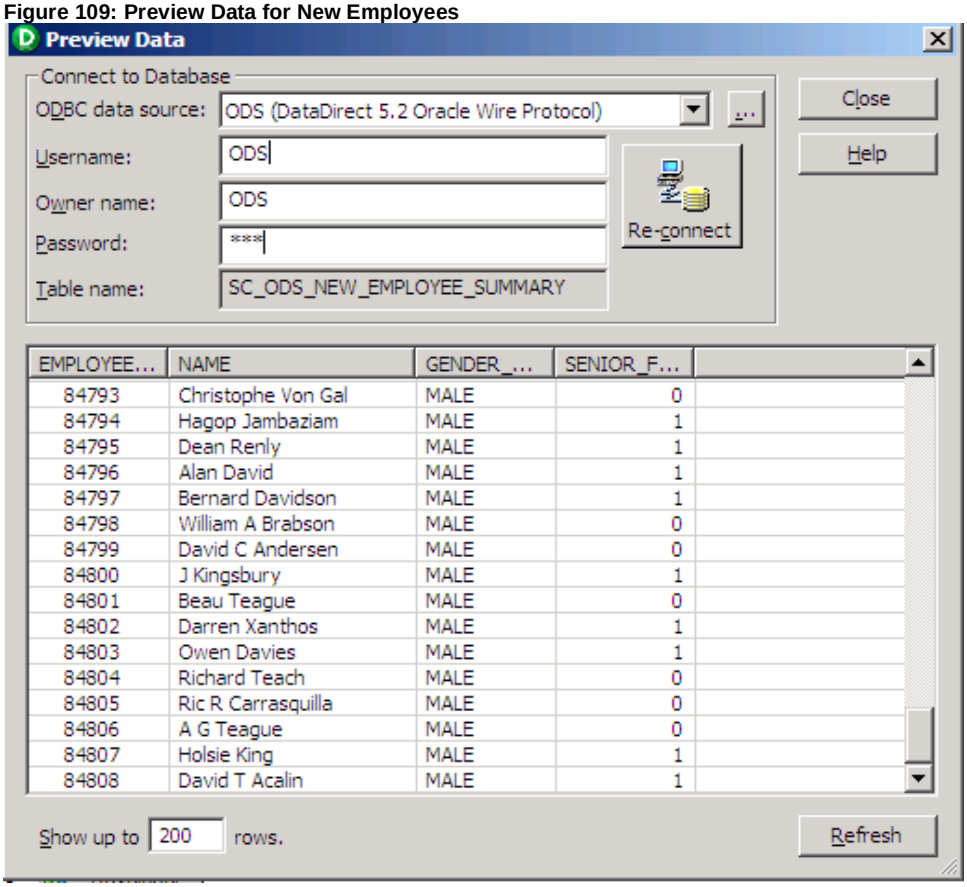
Step 4. Verify the Mapping Variable Incremented Properly

1) Preview the Target table.

a) Set the Show up to field to 200, to ensure that all 109 rows are visible.

b) Scroll to the bottom.

Figure 109: Preview Data for New Employees

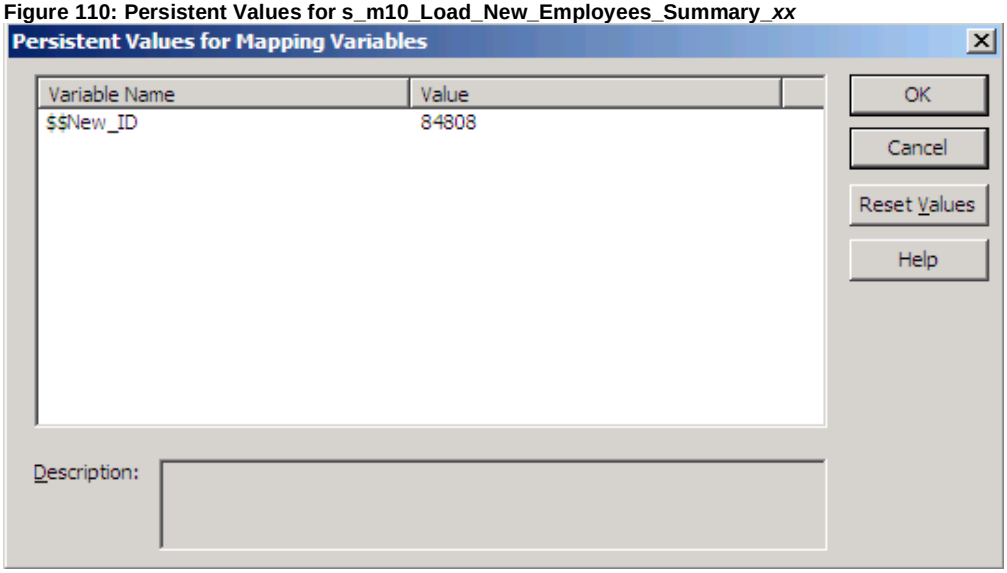


c) The last employee ID showing should be 84808.

2) Change to the Workflow Manager

3) Right-click the Session object and select View Persistent Values.

Figure 110: Persistent Values for s_m10_Load_New_Employees_Summary_*xx*



a) Note that the value of $$New_ID is the same as the value of the last EMPLOYEE_ID. It is ready for the next run of the workflow.

Step 5. Reconfigure and Rerun the Workflow

The purpose here is to verify that the counter is working properly.

1) Change the Source file to **employees_new2.dat**
2) Save and start the Workflow.
3) Viewing the Source/Target statistics, note that the Source file contained 5 rows that were added to the Target.
4) View the Persistent Values for the Session and verify that the number has incremented by five.
5) Preview the data in the Target and verify that five new employees have been added with the appropriate Employee ID numbers.

Extra Credit

If the Mapping had a relational source, how could a similar technique be used to read the Source incrementally, so that only new records would be read each time the Session was run?

Extra Credit Answer

If the Mapping had a relational source, how could a similar technique be used to read the Source incrementally, so that only new records would be read each time the Session was run?

You can use a Variable to hold a date/time stamp, which you would put into a SQL override and filter all records prior to that date/time.

A similar technique can be used to capture other ascending numbers such as DI, Oracle rownum, or SQL Identity fields.

Lab 11-1: Creating a Mapplet

Scenario:

- The team lead has noticed that there are situations where some of the transformations you developed in m8_Sales_Summaries_*xx* can be reused.
- To take advantage of these previously created objects, you will create a Mapplet from existing objects, which can then be used in other Mappings.

Goals:

- Create a Mapplet

Duration:

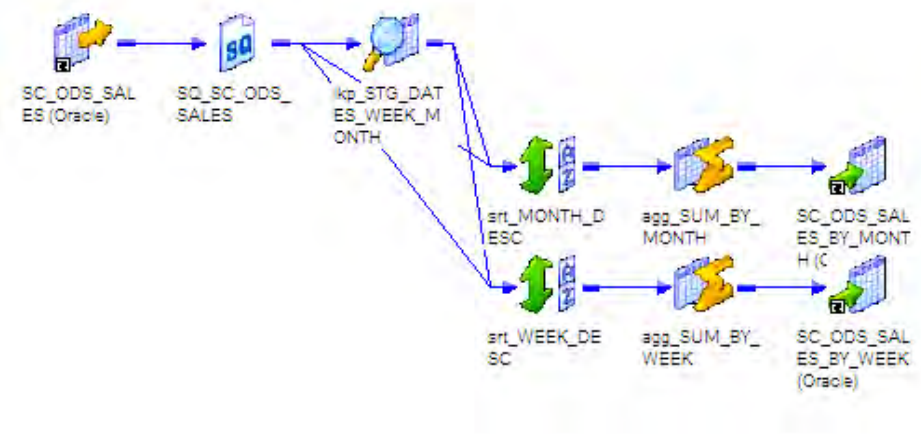
40 minutes

Instructions

Step 1. Copy and Prepare a Mapping

1) Make a copy of the Mapping m7_Sales_Summaries_*xx*.
2) Name it **m11_Sales_Summaries_*xx***.
3) Open the Mapping m11_Sales_Summaries_*xx*.
4) Arrange All Iconic if the Mapping isn't already arranged that way

Figure 111: Mapping m11_Sales_Summaries



Step 2. Create a Mapplet

1) Select the following transformations:
   lkp_STG_DATES_WEEK_MONTH
   srt_MONTH_DESC
   srt_WEEK_DESC
   agg_SUM_BY_MONTH
   agg_SUM_BY_WEEK
2) From the menu, select **Edit→Copy**.

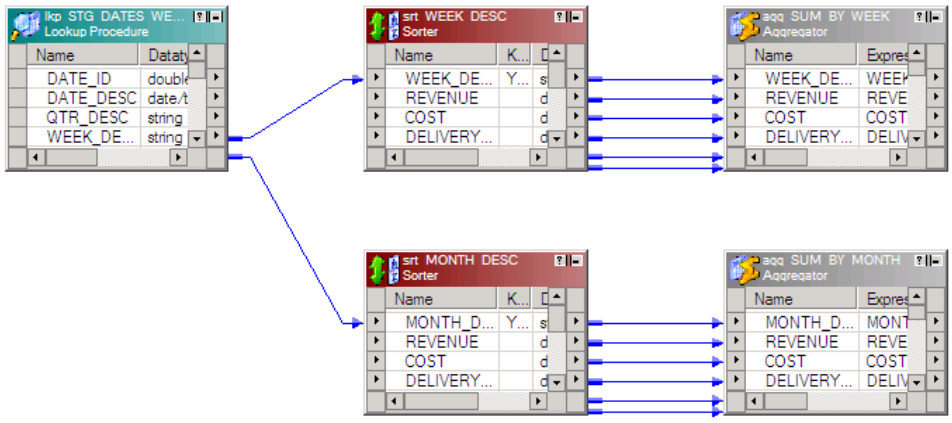
3) Open the Mapplet Designer tool ( ).

4) From the menu, select Edit→Paste.

5) Name the new Mapplet **mplt_Sales_Summaries**.

6) In the workspace, Arrange All.

Figure 112: New Mapplet



7) Add a Mapplet Input transformation ( ).

8) Drag ports to the Mapplet Input transformation:

   a) Drag the SALE_DATE port from the Lookup transformation to the Mapplet Input transformation.

   b) Drag every input port ***except*** WEEK_DESC from srt_WEEK_DESC to the Mapplet Input transformation.

   c) Drag every input port ***except*** MONTH_DESC from srt_MONTH_DESC to the Mapplet Input transformation.

9) Edit the Mapplet Input transformation.

   a) Rename it **in_Sales_Summaries**.

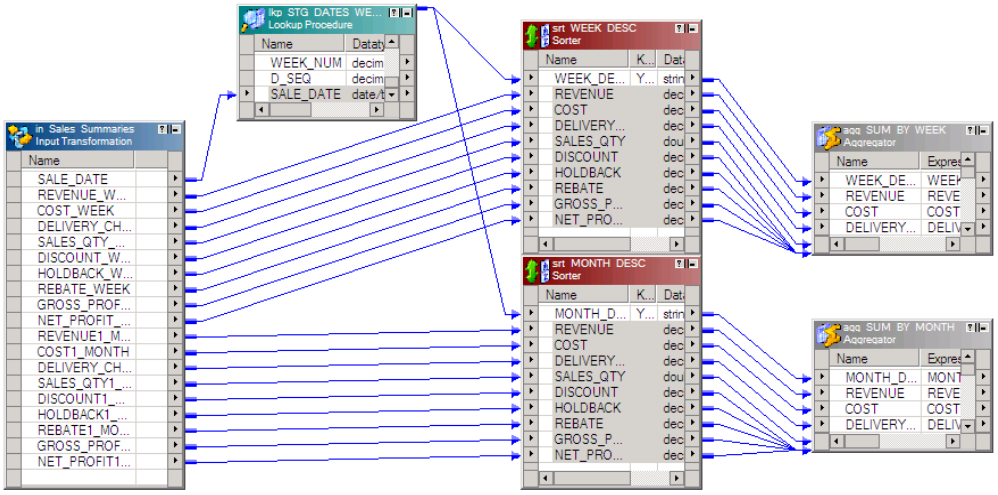
   b) To ports REVENUE through NET_PROFIT, add the suffix **_WEEK**.

Hint: You can use your mouse and Ctrl+C (copy) and Ctrl+V (Paste) to speed the process.

   c) On ports REVENUE1 through NET_PROFIT1, replace the 1 with the suffix **_MONTH**.

   d) Click **OK** to exit the Edit Transformations dialogue.

Figure 113: Mapplet with Input Transformation Added



10) Add a Mapplet Output transformation ( ) to the Mapplet.

   a) Rename it **out_Sales_Summary_Weekly**.

   b) From agg_SUM_BY_WEEK, drag ports WEEK_DESC and REVENUE_WEEK_SUM through NET_PROFIT_WEEK_SUM to the Mapplet Output transformation.

11) Add a second Mapplet Output transformation to the Mapplet.

   a) Rename it **out_Sales_Summary_Monthly**.

   b) From agg_SUM_BY_MONTH, drag ports MONTH_DESC and REVENUE_MONTH_SUM through NET_PROFIT_MONTH_SUM to the Mapplet Output transformation.

12) Save the Mapplet. Make sure it is valid.

Figure 114: Completed Mapplet

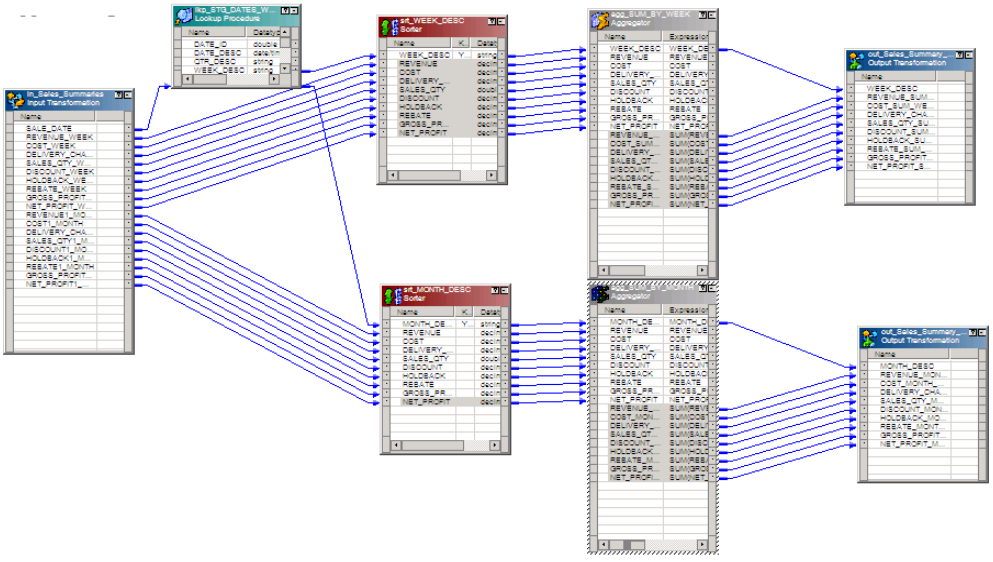
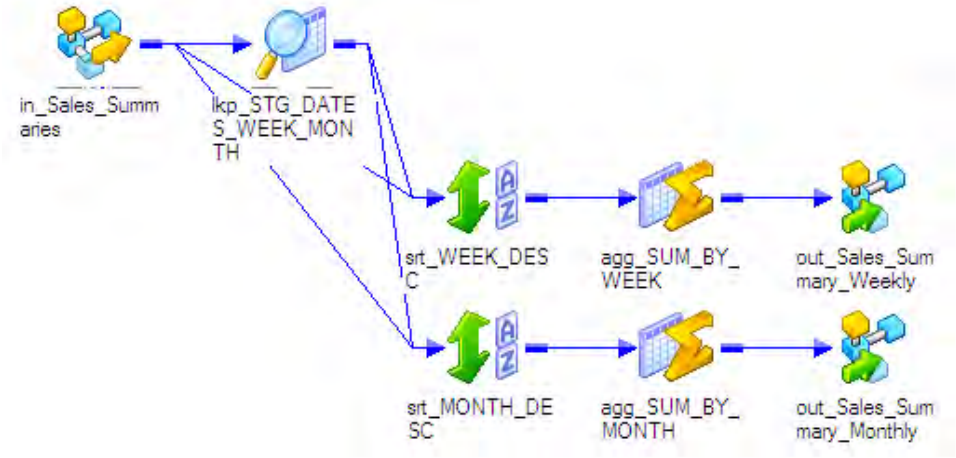


Figure 115: Mapplet Arranged Iconic



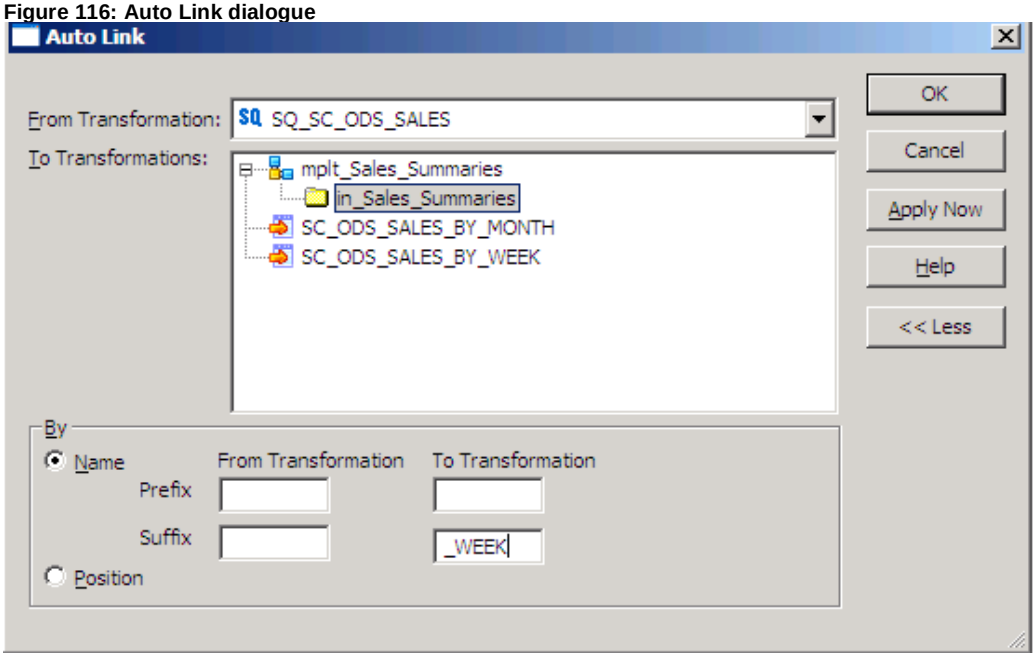
Step 3. Add the Mapplet to the Mapping

1) Switch to the Mapping Designer, where m11_Sales_Summary should still be open.
2) Delete the following transformations from the Mapping:
   lkp_STG_DATES_WEEK_MONTH
   srt_WEEK_DESC
   srt_MONTH_DESC
   agg_SUM_BY_WEEK
   agg_SUM_BY_MONTH
3) Drag the Mapplet **mplt_Sales_Summaries** into the mapping.
4) Manually link from the port SALE_DATE in the Source Qualifier to the port SALE_DATE in the in_SALES_SUMMARIES section of the Mapplet.

Hint: You may want to stretch the Mapplet vertically to see as many ports as possible.

5) You will use Autolink to connect the remaining input ports in the Mapplet.
   a) Right-click in the workspace and select **Autolink**.
   b) In the From Transformation dropdown box, select the Source Qualifier.
   c) In the To Transformation box, if necessary expand the Mapplet, and select the Input section in_Sales_Summaries.
   d) Click **More**.
   e) Enter **_WEEK** in the Suffix field under the To Transformation column, as shown:

Figure 116: Auto Link dialogue

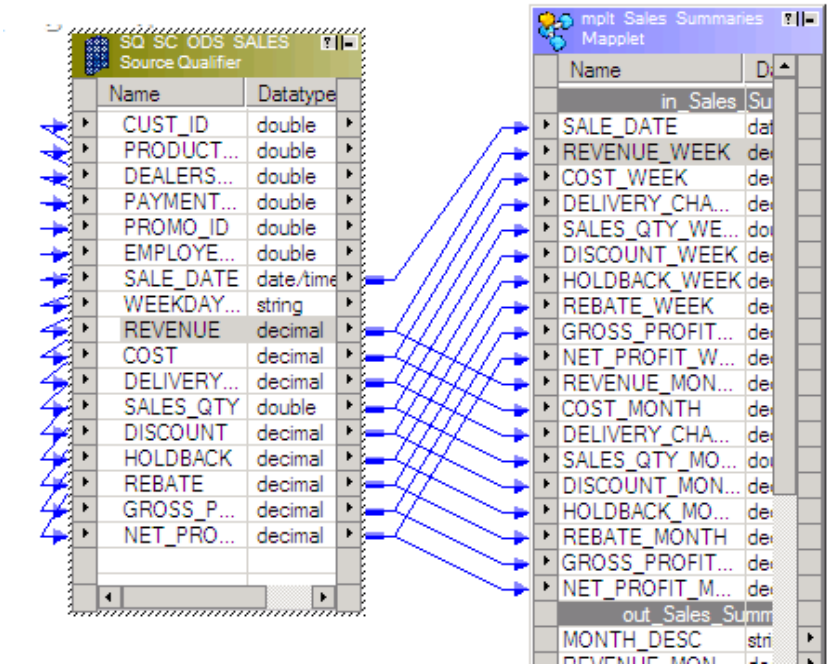


f) Click **OK**.

g) Half the ports are linked.

h) Repeat the process with the suffix **_MONTH** to complete the links to the Input section of the Mapplet.

Figure 117: SQ Linked to Mapplet



6) Now you will link the Mapplet to the Targets.

a) In the Out_Sales_Summary_Monthly section, click MONTH_DESC.

b) Hold down the SHIFT button on the keyboard and click NET_PROFIT_MONTH_SUM to select all the fields in the section.

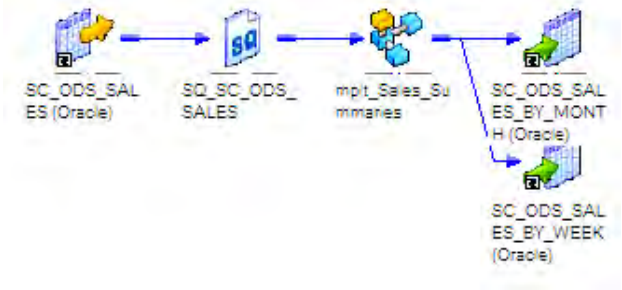
c) Drag the mouse from MONTH_DESC to MONTH_DESC in the Target definition SC_ODS_SALES_BY_MONTH.

Tip: This can work because the ports are in exactly the same order on both the Mapplet output section and the Target. It is equivalent to Autolink by Position, but does not automatically start with the first port on each transformation.

d) Using the same procedure, link the ports of the Out_Sales_Summary_Weekly to the Target SC_ODS_SALES_BY_WEEK.

7) Save the Mapping and make sure it is valid.

Figure 118: m11_Sales_Summaries_*xx*



Lab 12-1: Using Assignment and Decision Tasks

Scenario:

- Management wants the ability to report sales on a weekly basis.
- You created a Mapping, (**m6_Load_ODS_SALES_*xx***), that enables this reporting.
- You need to put this Mapping into production with a more formal and fault-tolerant Workflow.
- Data must be retrieved from the OLTP source.

Goals:

- Create a more formal Workflow that prevents some types of bad data from getting into the ODS_SALES table.
- Assign Workflow variables to keep track of the number of times the Workflow has been run.
- Increment Workflow variables using an Assignment task.
- Branch in a Workflow using link conditions and a Decision task to choose to run the next Session or report an error.

Duration:

45 minutes

Instructions

Step 1. Make a Session Reusable and Copy a Reusable Session

1) Make a Session Reusable

   a) In the Workflow Manager application, locate the Workflow in your folder named **wkf_Load_ODS_SALES_*xx*** and drag it into the Workflow Designer workspace.

   b) Edit the Session

      (i) In the General tab, check the "Make reusable" checkbox.

      (ii) Click **Yes** to make the Session reusable.

      (iii) Click **OK**.

   c) Save the Workflow.

   d) Disconnect from your ~Devloper*xx* folder.

2) Use the R button () to switch to the Repository Manager application.

   a) Open your ~Developer*xx* folder.

   b) Open the SC_DATA_STRUCTURES folder.

   c) Using the Edit→Copy technique shown in an earlier lab, copy the reusable Session **s_m_Load_STG_TRANSACTIONS** from SC_DATA_STRUCTURES to your ~Developer*xx* folder.

   d) Disconnect from the repository. (The Repository Manager automatically saves your folder.)

3) Switch back to the Workflow Manager application.

   a) Open your folder.

   b) Verify that there are two reusable Sessions in the Sessions subfolder: s_m_Load_STG_TRANSACTIONS and s_m8_Load_ODS_SALES_*xx*.

   c) Select the Task Developer tool

      (i) Drag the Session s_m_Load_STG_TRANSACTIONS into the workspace.

      (ii) Edit the Session and change its name by adding the suffix *_xx*.

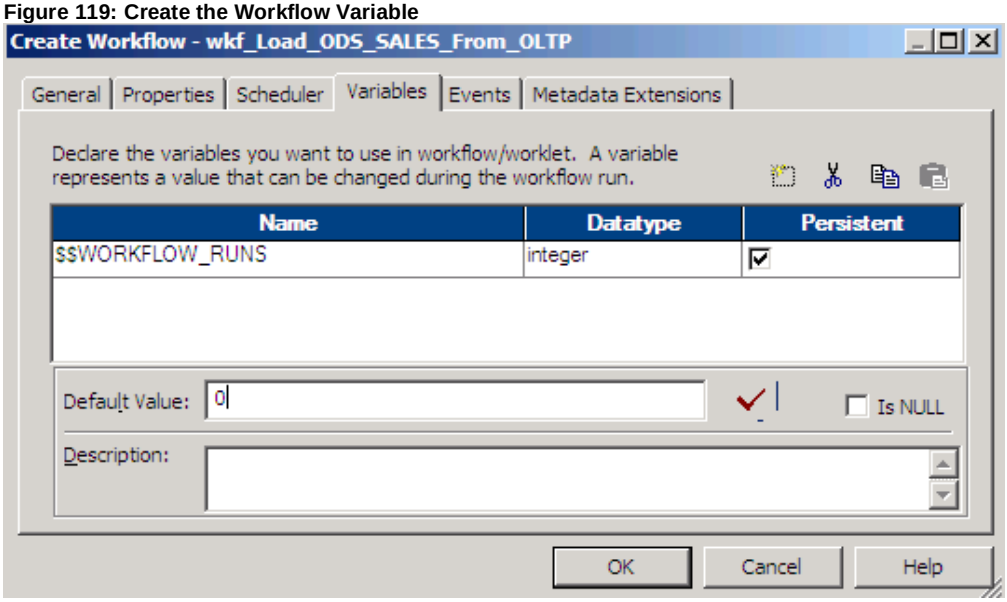
Step 2. Create and Configure the Workflow

1) Create a new Workflow named **wkf_Load_ODS_SALES_from_OLTP_xx**

2) In the Create Workflow dialogue, select the Variables tab.

   a) Create a new Workflow variable with the following properties:
   - Variable Name =**$$WORKFLOW_RUNS**
   - Datatype = **integer**
   - Persistent = **checked**
   - Default value = **0**

Figure 119: Create the Workflow Variable



3) Drag the Session **s_m_Load_STG_TRANSACTIONS_*xx*** into the workflow and link the Start task to it.

4) Set the Relational reader for the source to SDBU.

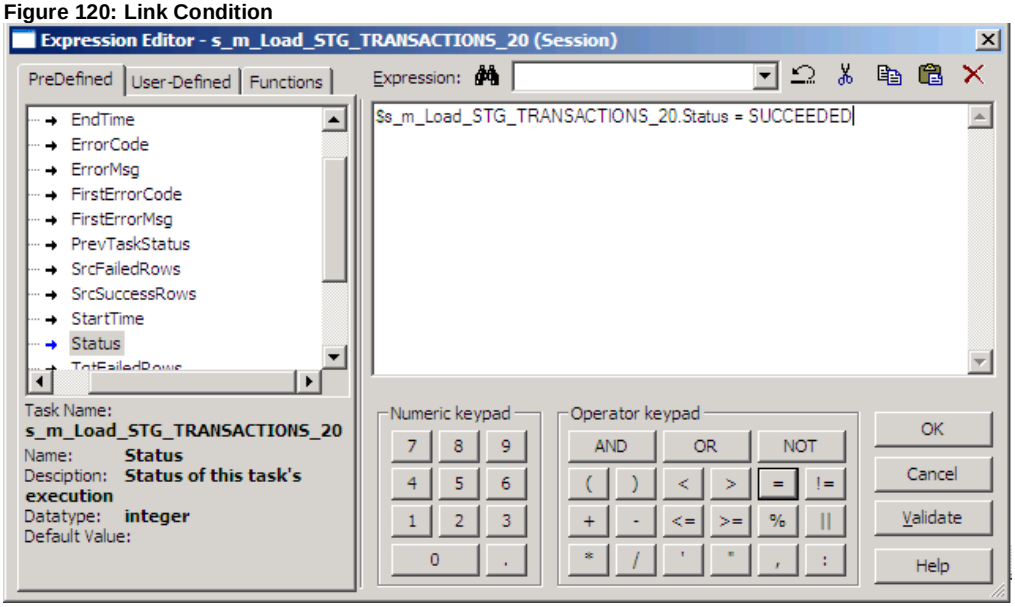
5) Set the Relational writer for the target to STG*xx*.

6) Save your work.

Step 3. Create the Assignment Task

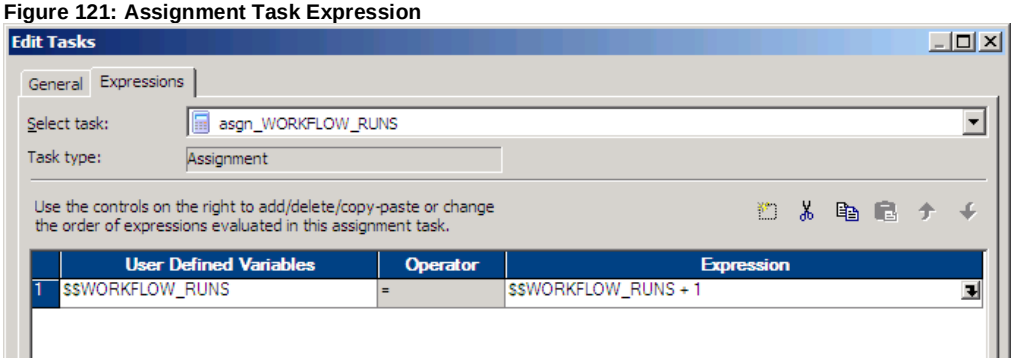
1) Use the Assignment Task button ( ) to add an Assignment task to the Workflow.
2) Link the Session task to the Assignment task.
3) Double-click the link to edit it.
4) Add a link condition to ensure that the Assignment task executes only if the Session task was successful.
   a) Select the pre-defined function "Status" and set the condition so that the status must be SUCCEEDED. (See figure.)

Figure 120: Link Condition



5) Edit the Assignment task.
   a) Change its name to **asgn_WORKFLOW_RUNS**.
   b) In the Expressions tab, create an expression that increments the User Defined Variable by one. (See figure.)

Figure 121: Assignment Task Expression

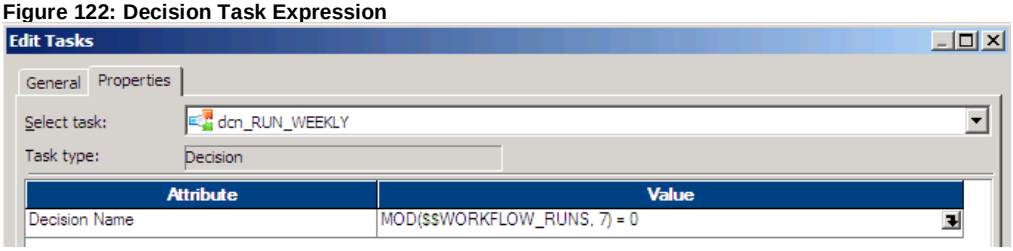


6) Save your work.

Step 4. Create the Decision Task

1) Use the Decision task button ( ) to add a Decision task to the Workflow.
2) Link asgn_WORKFLOW_RUNS to the Decision task.
3) Edit the link.
   a) Add a condition to ensure that the Decision task executes only if assgn_WORKFLOW_RUNS completed successfully.
4) Edit the Decision task
   a) Rename it **dcn_RUN_WEEKLY**.
   b) In the Properties tab, create a Decision Name expression to see if this is the seventh day of the Workflow week.
      (i) The Modulus function (MOD) divides two numbers and yields the remainder.
      (ii) See the figure.

Figure 122: Decision Task Expression



Tip: The decision task evaluates an expression and returns a value of either TRUE or FALSE. This value can be checked in a Link condition to determine the direction in which the Workflow proceeds from the Decision task.

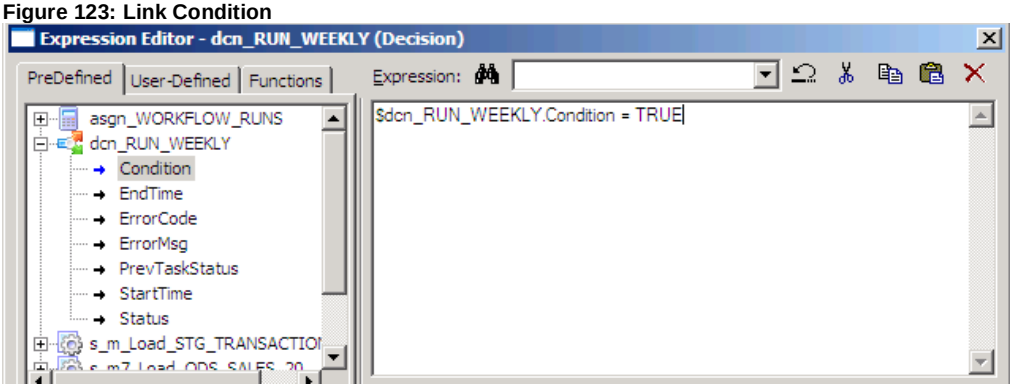
5) Save your work.

Step 5. Add a Second Session

1) Drag the session **s_m6_Load_ODS_SALES_*xx*** into the Workflow.
2) Link dcn_RUN_WEEKLY to it.
3) Double-click the link.

a) Add a link condition that checks whether dcn_RUN_WEEKLY has returned a value of TRUE. (See figure.)

Figure 123: Link Condition

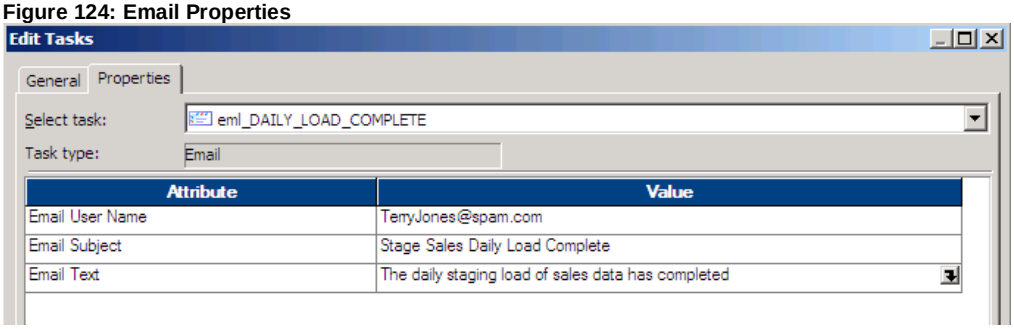


4) The Session properties were set correctly in the Workflow where you first created this Session.

Step 6. Create the Email Task

1) Use the email task button ( ) to add an Email task to the mapping.
2) Link dcn_RUN_WEEKLY to the Email task.
3) Add a Link condition that checks whether dcn_RUN_WEEKLY has returned a value of FALSE.
4) Edit the Email task.
   a) Rename it **eml_DAILY_LOAD_COMPLETE**
   b) In the Properties tab, enter appropriate values for Email User Name, Email Subject, and Email Text (see example below).

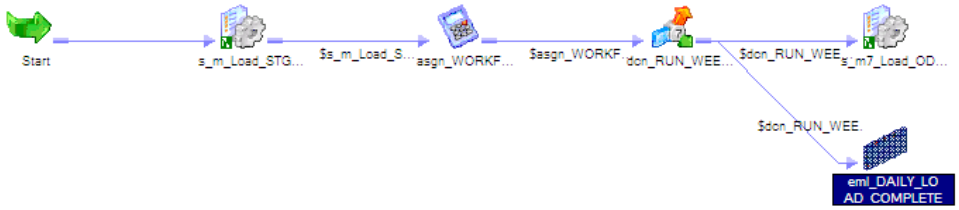
Figure 124: Email Properties



c) Click **OK**.

d) Right-click in the workspace and select **Arrange → Horizontal**

Figure 125: wkf_Load_ODS_SALES_From_OLTP completed



e) Save your work.

Step 7. Start the Workflow and Monitor the Results

You will need to run the Workflow seven times in order to test the weekly aggregate session.

1) Start the Workflow.

a) Review the Workflow results in the Gantt view of the Workflow Monitor. It should appear similar to the figure below:

Figure 126: Gantt Chart of the Completed Workflow Run

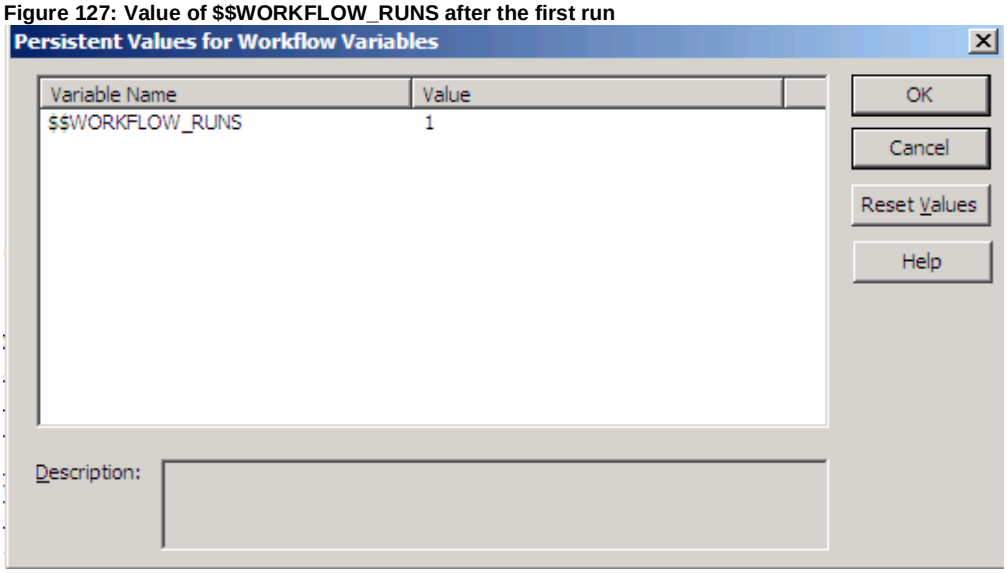
Name	Duration	Status	3:00pm	4:00pm	5:00pm
Repositories					
REP_SVC_EDW_DEV					
INT_SVC_EDW_DEV	15:06:07	Connected			
EDW					
SC_DATA_STRUCTURES					
~Developer19					
~Developer20					
wkf_Load_ODS_SALES_From_OLTP	00:00:16	Succeeded			
s_m_Load_STG_TRANSACTION...	00:00:15	Succeeded			
asgn_WORKFLOW_RUNS	00:00:00	Succeeded			
dcn_RUN_WEEKLY	00:00:00	Succeeded			

2) Return to the Workflow Manager.

3) In the Navigator window, in the Workflows subfolder, right-click **wkf_Load_ODS_SALES_From_OLTP_*xx*** and select **View Persistent Values.**

   a) The value of $$WORKFLOW_RUNS should be 1.

Figure 127: Value of $$WORKFLOW_RUNS after the first run



4) Run the workflow six more times to simulate a week's normal runs.

5) After the last run, how is the Gantt chart different?

Extra Credit:

Modify the Workflow to fail if any of the Sessions in the Workflow fail.

Hint: You will need to use more than one Control task.

Hint: You can force a Session failure by changing to a Relational connection that references a database schema that does not have the table in it. For example, change the target table to use Relational connection OLTP.

Answers

7.5. After the last run, how is the Gantt char different?
The second Session task is shown connected to the Decision task, and has a status of Succeeded.

Lab 12-2: Using Events and Timers

Scenario:

- The sales summary tables can only be loaded after the ODS_SALES table is loaded.
- Management has further determined that if the workflow takes more than 15 minutes to run, you should fail the workflow as something has probably gone wrong.
- You created a Mapping, (**m7_Load_SALES_SUMMARIES**), that loads the Sales Summary tables.
- You created a Mapping (**m6_Load_ODS_Sales**) that loads the ODS_Sales table.
- Data must be retrieved from and written to the ODS database schema.

Goals:

- Create a Workflow that loads the ODS_SALES table, then raises an User-Defined event.
- Wait for the User-Defined event, then load the Sales Summaries tables.
- Stop the workflow nicely if the Sales Summary tables load properly.
- Create a third branch to the workflow that starts a 15-minute timer. If the time limit is reached, then fail the workflow.
- Set the workflow to run at a particular time.

Duration:

35 minutes

Instructions

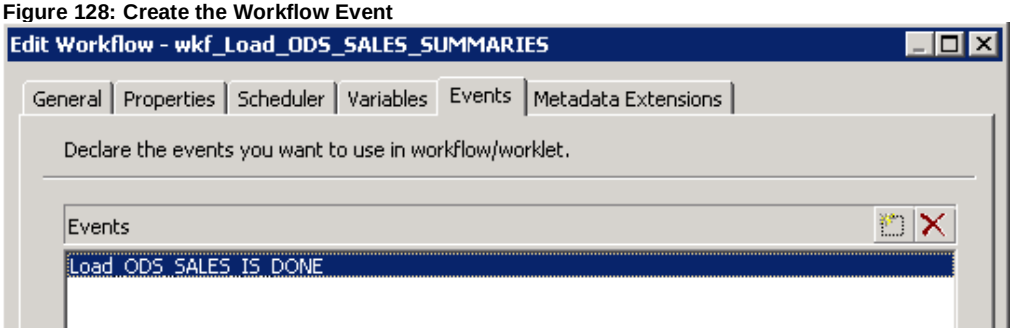
Step 1. Make a Session Reusable

1) In the Workflow Manager application, locate the Workflow in your folder named **wkf_Load_SALES_SUMMARIES_*xx*** and drag it into the Workflow Designer workspace.
   a) Edit the Session
      (i) In the General tab, check the "Make reusable" checkbox.
      (ii) Click **Yes** to make the Session reusable.
      (iii) Click **OK**.
   b) Save the Workflow.

Step 2. Create and Configure the Workflow

1) Create a new Workflow named **wkf_Load_ODS_SALES_SUMMARIES**
2) In the Create Workflow dialogue, select the Events tab.
   - Create a new Event named **Load_ODS_SALES_IS_DONE**

Figure 128: Create the Workflow Event



3) Drag the Session **s_m8_Load_ODS_Sales** into the workflow and link the Start task to it.

4) Save your work.

Step 3. Create the Event Raise Task

1) Use the Event Raise Task button ( ) to add an Event Raise task to the Workflow.

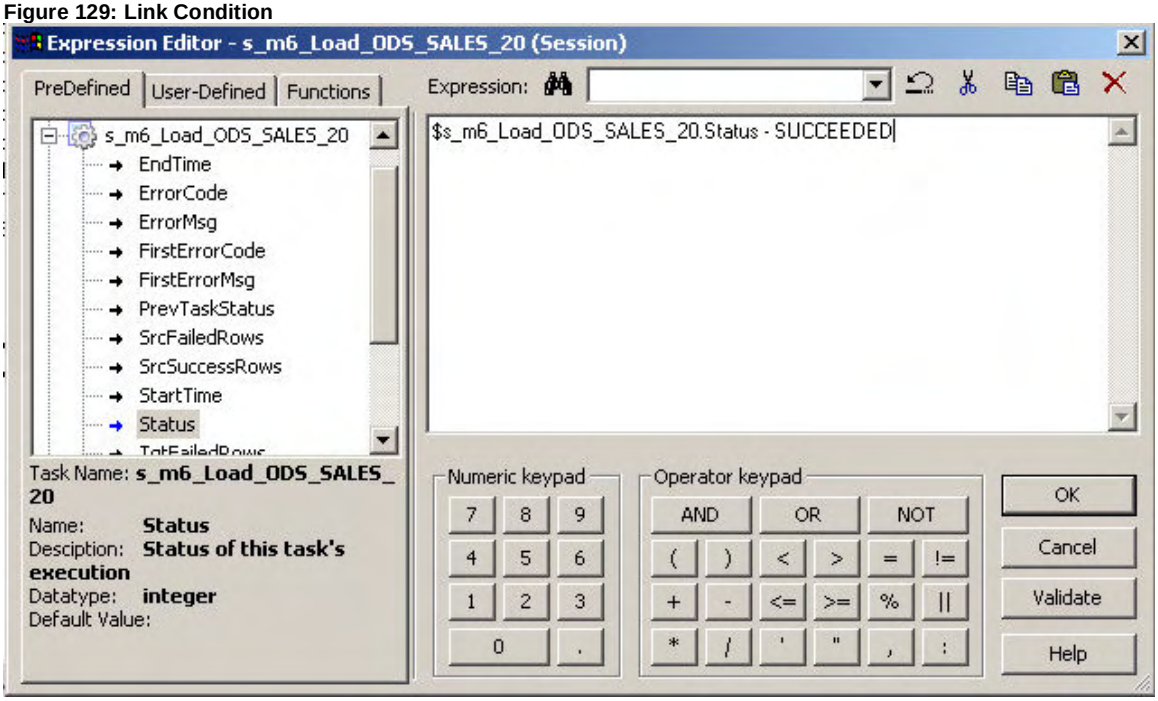
2) Link the Session task to the Event Raise task.

3) Double-click the link to edit it.

4) Add a link condition to ensure that the Event Raise task executes only if the Session task was successful.

   a) Double-click the pre-defined function "Status" and set the condition so that the status must be SUCCEEDED. (See figure.)

Figure 129: Link Condition

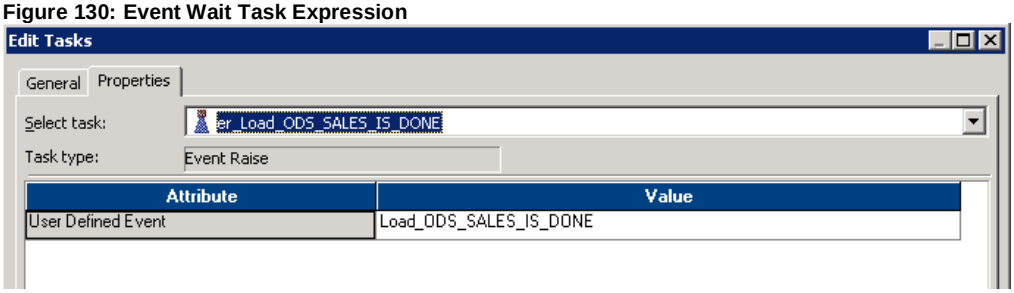


5) Edit the Event Raise task.

   a) Change its name to **er_Load_ODS_SALES_IS_DONE**.

   b) In the Properties tab, set the User Defined Event to wait for. (See figure.)

Figure 130: Event Wait Task Expression

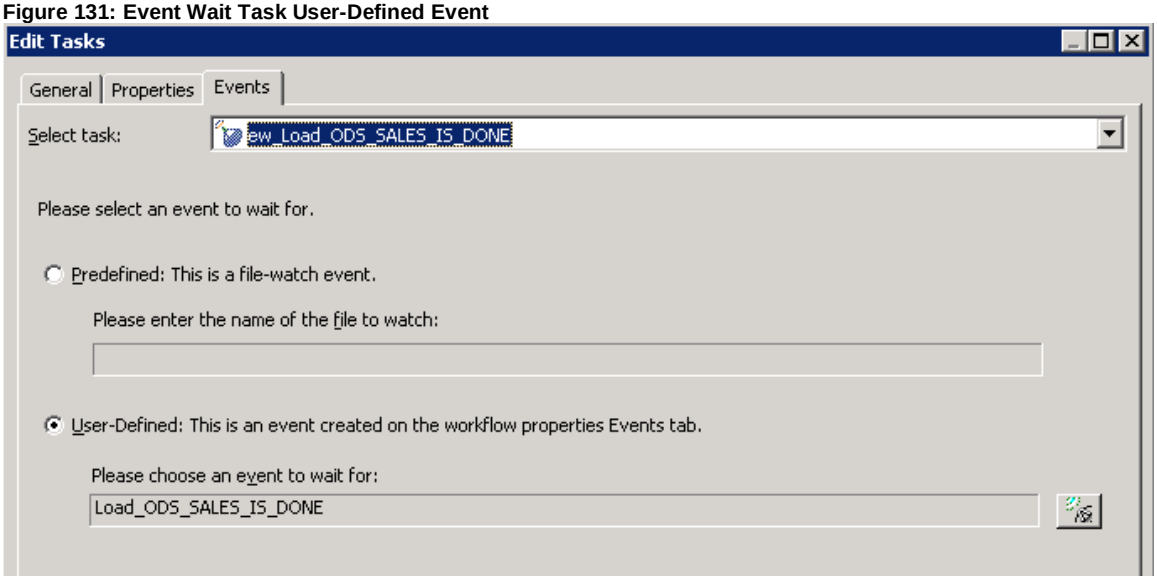


6) Save your work.

Step 4. Create the Event Wait Task

1) Use the Event Wait task button ( ) to add an Event Wait task to the Workflow.
2) Link the Start task to the Event Wait task.
3) Edit the Event Wait task
   a) Rename it **ew_Load_ODS_SALES_IS_DONE**.
   b) In the Events tab, set a User-Defined event which the Event Wait task will wait for before executing. See the figure.

Figure 131: Event Wait Task User-Defined Event



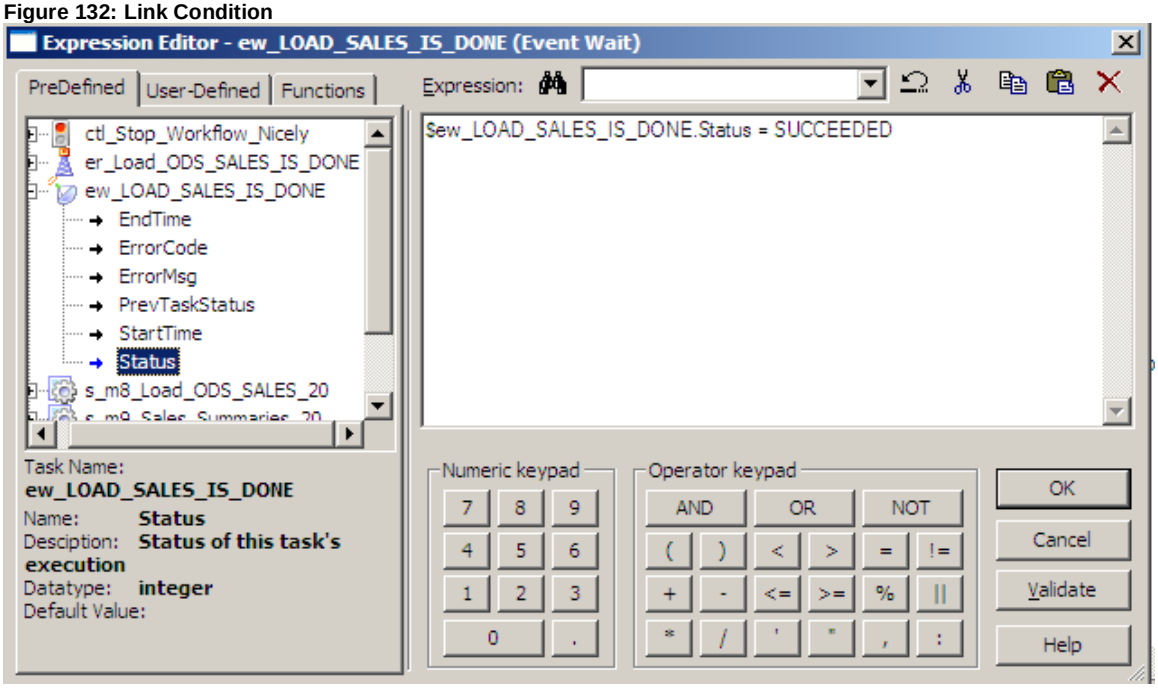
4) Save your work.

Step 5. Add a Second Session

1) Drag the session **s_m7_Sales_Summaries_*xx*** into the Workflow.
2) Link ew_Load_ODS_SALES_IS_DONE to it.
3) Double-click the link.

a) Add a link condition that checks whether the Event Wait task has completed successfully. (See figure.)

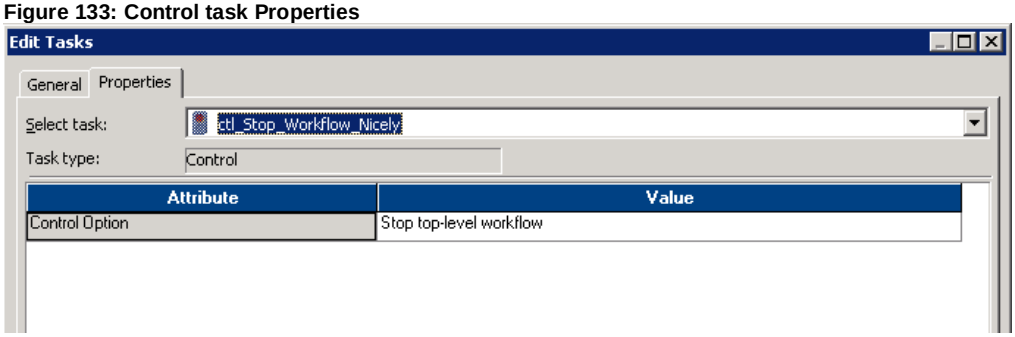
Figure 132: Link Condition



Step 6. Create the first Control Task

1) Use the control task button ( ) to add a Control task to the workflow.
2) Link s_m7_Sales_Summaries to the Control task.
3) Add a Link condition that checks whether s_m7_Sales_Summaries has returned a status of SUCCEEDED.
4) Edit the Control task.
   a) Rename it **ctl_Stop_Workflow_Nicely**
   b) In the Properties tab tell the Control task to stop the top-level workflow (see example below).

Figure 133: Control task Properties

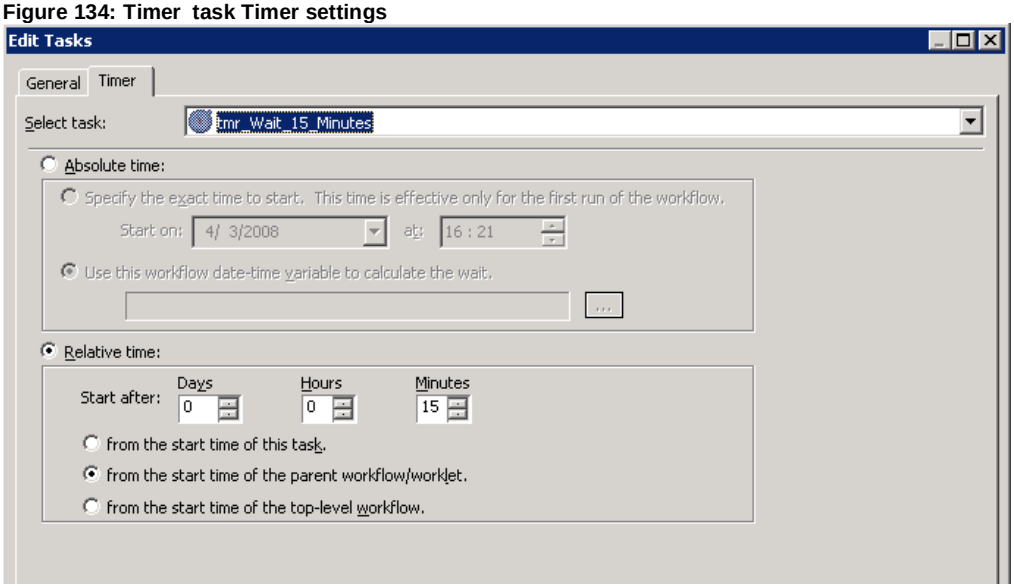


   c) Click **OK**.

Step 7. Add a Timer task

1) Use the Timer task button ( ) to add a Timer task to the workflow.
2) Link the Start task to it.
3) Edit the Timer task.
   a) Rename it **tmr_Wait_15_Minutes**
   b) In the Timer tab tell the Timer task to count 15 minutes from the time the parent workflow started (see example below).

Figure 134: Timer task Timer settings

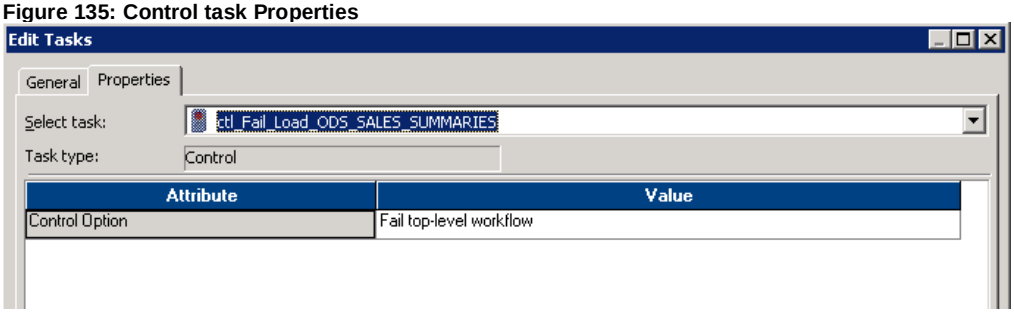


   c) Click **OK**

Step 8. Create the Second Control Task

1) Use the control task button ( ) to add a second Control task to the workflow.
2) Link tmr_Wait_15_Minutes to the Control task.
3) Add a Link condition that checks whether tmr_Wait_15_Minutes has returned a status of SUCCEEDED.
4) Edit the Control task.
   a) Rename it **ctl_Fail_Load_ODS_SALES_SUMMARIES**
   b) In the Properties tab tell the Control task to Fail the top-level workflow (see example below).

Figure 135: Control task Properties

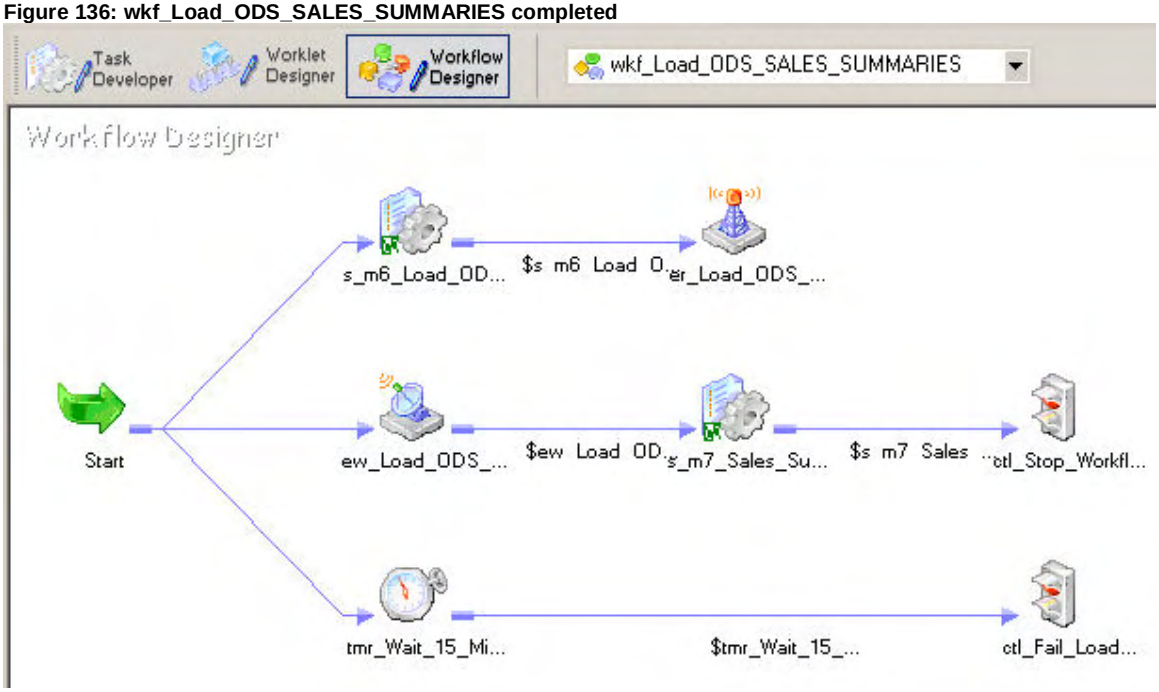


c) Click **OK**.

Step 9. Verify the workflow

1) Save your work.
2) Verify the workflow is valid
3) Right-click in the workspace and select **Arrange → Horizontal**

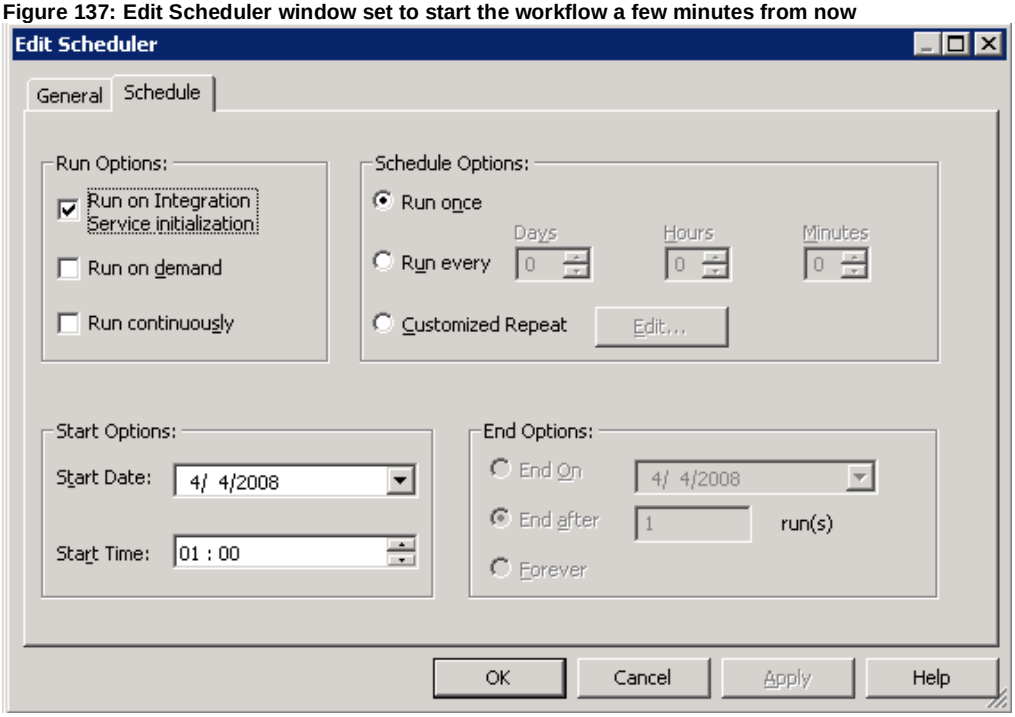
Figure 136: wkf_Load_ODS_SALES_SUMMARIES completed



Step 10. Use the Scheduler to set the Workflow to Start at a Given time

1) Edit the workflow
2) Select the Scheduler tab
3) Click the Scheduler button ()
4) In the Edit Scheduler windows, select the Schedule tab
5) Check the **Run on Integration Service Initialization** check box
6) Set the workflow to start a few minutes from now. For example, if it is 12:55AM, set the workflow to start at 1:00AM. (see figure below)

Figure 137: Edit Scheduler window set to start the workflow a few minutes from now



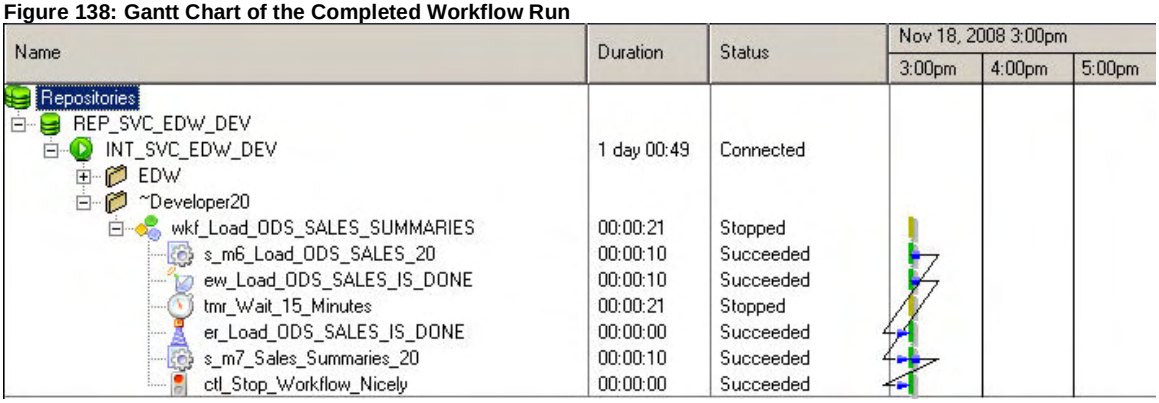
7) Click **OK**
8) Click **OK**
9) Save your work

Step 11. Start the Workflow and Monitor the Results

1) The workflow will start, but not execute until the date and time set in the Scheduler. Wait until it starts.

2) Review the Workflow results in the Gantt view of the Workflow Monitor. It should appear similar to the figure below:

Figure 138: Gantt Chart of the Completed Workflow Run



Note that the first Control task stopped the workflow before the second one failed it. The first Control task is needed so the second one doesn't execute after 15 minutes every time the workflow is run.

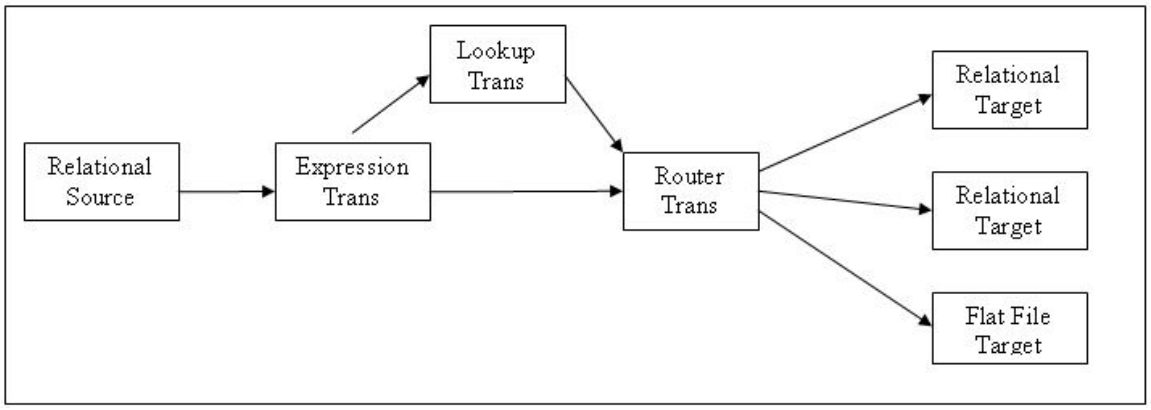
Workshop 1: Mapping Design Workshop

What to Consider

The mapping process requires much more up-front research than it may appear. Before designing a Mapping, it is important to have a clear picture of the end-to-end processes that the data will flow through.

- Design a high-level view of the mapping and document a picture of the process with the mapping, using a textual description to explain exactly what the Mapping is supposed to accomplish and the methods or steps it will follow to accomplish its goal.

Figure 139: Example of a high-level process overview



- After the high-level flow has been established, document the details at the field level, listing each of the Target fields and the Source field(s) used to create each Target field.
  - Document any expression that may be needed to generate the Target field (e.g.: a sum of a field, a multiplication of two fields, a comparison of two fields, etc.).
  - Whatever the rules, be sure to document them at this point, and remember to keep it at a physical level.
  - The designer may have to do some investigation at this point for some business rules. For example, the business rules may say "For active customers, calculate a late fee rate." The designer of the Mapping must determine that, on a physical level, this translates to "for customers with an ACTIVE_FLAG of 1, multiply the DAYS_LATE field by the LATE_DAY_RATE field."
- Create an inventory of Mappings and reusable objects. This list is a "work in progress" and will have to be continually updated as the project moves forward.
  - These lists are valuable to everyone, but especially for the lead developer. These objects can be assigned to individual developers and progress tracked over the course of the project.
- The administrator or lead developer should gather all the potential Sources, Targets, and reusable objects and place them in a folder accessible to all who may need access to them.
  - In our labs, this has been SC_DATA_STRUCTURES.
- If a shared folder for Sources and Targets is not available, the developer will need to obtain the Source and Target database schema owners, passwords, and connect strings.

  - With this information, ODBC connections can be created in the Designer tool to allow access to the Source and Target definitions.
  - This information will also be needed to create Connections in the Workflow Manager.

- Reusable objects need to be properly documented to make it easier for other developers to determine whether they can/should use them in their own development.
- A developer's specifications for a Mapping should include, at a minimum, the required Sources and Targets, and any additional information regarding derived ports, and how the ports relate from the Source to the Target.
- The Informatica Velocity methodology provides a matrix that assists in detailing the relationships between Source fields and Target fields (Mapping Specifications.doc). It also depicts fields that are derived from values in the Source and eventually linked to ports in the Target.
- Document any other information about the Mapping that is likely to be helpful in developing it. Helpful information may, for example, include Source and Target connection information, Lookups (and how to match data in the Lookup table), potential data issues at a field level, any known issues with particular fields, pre-or post-Mapping processing requirements, and any information about specific error handling requirements for the Mapping.
- The completed Mapping design should then be reviewed with one or more team members for completeness and adherence to the business requirements.
  - In addition, the design document should be updated whenever the business rules change, or if more information is gathered during the build process.

Mapping Specifics

The following tips will make the Mapping development process more efficient. (Not in any particular order.)

- One of the first things to do is to bring all required Source and Target objects into the Mapping.
- Only connect fields that are needed or will be used.
  - Note, however, that all ports must be connected from the Source definition to the Source Qualifier transformation.
  - Only needed fields should be projected from Source Qualifiers that originate with Relational tables. The SQL that PowerCenter generates will include only the needed fields, reducing computing resource requirements. In this case, only connect from the Source Qualifier those fields that will be used subsequently.
- Filter rows early and often. Only manipulate data that needs to be moved and transformed. Reduce the number of non-essential records passed through the Mapping.
- Decide if a Source Qualifier join will net the result needed, versus creating a Lookup to retrieve desired results.
- Reduce the total number of transformations. Excessive number of transformations will increase overhead.

- Make use of variables (local or global) to reduce the number of times functions will have to be used.
- Watch your datatypes. The Informatica engine converts compatible datatypes automatically, but unnecessary conversion is inefficient.
- Make use of variables, reusable transformations, and Mapplets as "reusable code." These will leverage the work being done by others, promote standardization, and ease maintenance tasks.
- Use active transformations early in the process to reduce the number of records as early in the Mapping as possible.
- When joining Sources, select the appropriate master (driving) table.
- Utilize single-pass reads. Design Mappings to utilize one Source Qualifier to populate multiple Targets.
- Remove or reduce field-level stored procedures.
    - Even though PowerCenter does a lot to increase efficiency, Stored Procedure objects will be executed for each record, and slow performance.
- Lookup transformation tips:
    - When the Source is large, cache lookup tables columns for lookup tables with 500,000 rows or less on 32-bit platforms with limited RAM memory.
    - Standard rule of thumb is not to cache tables over 500,000 rows on 32-bit platforms with limited RAM.
    - Use equality (=) conditions if possible in the Condition tab.
    - Use IIF or DECODE functions when lookup returns small rowsets.
    - Avoid data comparisons in lookup; convert to string.
- Operations and Expression transformation tips
    - Numeric operations are faster than string.
    - Trim Char and Varchar fields before performing comparisons.
    - Operators are faster than functions (i.e., || is faster than the CONCAT function).
    - Use flat files. File read/writes are faster than database read/writes on the same server. Fixed-width files are faster than delimited file processing.

Workshop

Scenario:

- Management wants the ability to analyze how certain promotions are performing. They want to be able to gather the promotions by day for each dealership, for each product sold.

Goals:

- Design and create a mapping to load the aggregate table.

Duration:

120 minutes

Instructions

Sources and Targets

Sources: TRANSACTIONS and PRODUCT_COST

These relational tables contain sales transactions and Product cost data for seven days. They are located in the SDBU schema. For the purpose of this mapping, we will read all the data in these tables.

These tables can be joined on PRODUCT_ID and PRODUCT_CODE

Figure 140: TRANSACTION table definition

TRANSACTIONS (Oracle)

K.	Name	Datatype	Length/Precision
	CUST_ID	number(p,s)	4
	PRODUCT_ID	number(p,s)	5
	DEALERSHIP_ID	number(p,s)	2
	PAYMENT_DESC	varchar2	10
	PROMO_ID	number(p,s)	3
	DATE_ID	number(p,s)	7
	TRANSACTION...	varchar2	29
	TRANSACTION...	number(p,s)	7
	EMPLOYEE_ID	number(p,s)	5
	TIME_KEY	number(p,s)	6
	SELLING_PRICE	number(p,s)	8
	DELIVERY_CH...	number(p,s)	7
	QUANTITY	number(p,s)	1
	DISCOUNT	number(p,s)	9
	HOLDBACK	number(p,s)	7
	REBATE	number(p,s)	7

Figure 141: PRODUCT_COST table definition

PRODUCT_COST (Oracle)

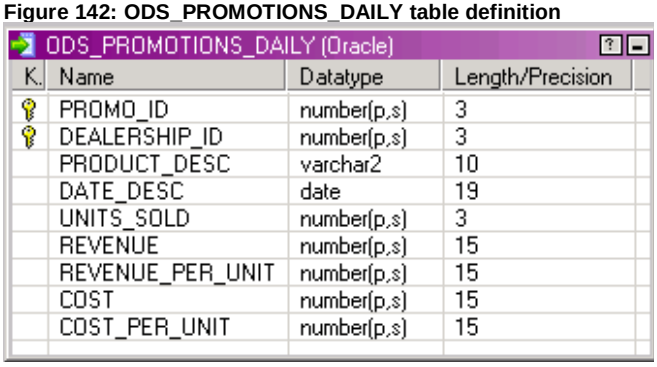
K.	Name	Datatype	Length/Precision
🔑	PRODUCT_CODE	number(p,s)	5
	SUPPLIER_DESC	varchar2	50
	COMPONENT_ID	varchar2	50
	COST	number(p,s)	9

A

Target: ODS_PROMOTIONS_DAILY

This is a relational table located in the ODS*xx* schema. After running the Mapping, it should contain 1283 rows.

Figure 142: ODS_PROMOTIONS_DAILY table definition



ODS_PROMOTIONS_DAILY (Oracle)

K.	Name	Datatype	Length/Precision
	PROMO_ID	number(p,s)	3
	DEALERSHIP_ID	number(p,s)	3
	PRODUCT_DESC	varchar2	10
	DATE_DESC	date	19
	UNITS_SOLD	number(p,s)	3
	REVENUE	number(p,s)	15
	REVENUE_PER_UNIT	number(p,s)	15
	COST	number(p,s)	15
	COST_PER_UNIT	number(p,s)	15

Mapping Details

In order to successfully create the mapping, you will need to know some additional details.

- Management has decided that they don't need to keep track of the Manager Discount and the Employee Discount (PROMO_ID 105 and 200), so these will need to be excluded from the load.
- The PRODUCT_DESC can be obtained from the PRODUCT table by matching on the PRODUCT_ID.
- The DATE_DESC can be obtained from the STG_DATES table by matching the TRANSACTION table DATE_ID to the DATE_ID in STG_DATES.
- UNITS_SOLD is derived by summing QUANTITY.
- REVENUE is derived by taking the value in the QUANTITY port times the SELLING_PRICE and then subtracting the DISCOUNT, HOLDBACK and REBATE.
    - Most of the discounts are valid but occasionally they may be higher than the acceptable value of 17.25%. When this occurs you will need to obtain an acceptable value based on the PROMO_ID. The acceptable value can be obtained from the PROMOTIONS table by matching the PROMO_ID.
    - The DISCOUNT is a percentage stored as a number. To calculate the actual discount in dollars, divide the DISCOUNT by 100 and multiply it by the SELLING_PRICE.
    - Revenue per unit is REVENUE divided by QUANTITY.
- COST is derived by summing UNIT_COST.
- COST_PER_UNIT is derived by summing the UNIT_COST and dividing it by the sum of QUANTITY.
- Data will be aggregated by PROMO_ID, DEALERSHIP_ID, PRODUCT _DESC, and DATE_DESC.
- Save your work often!!!

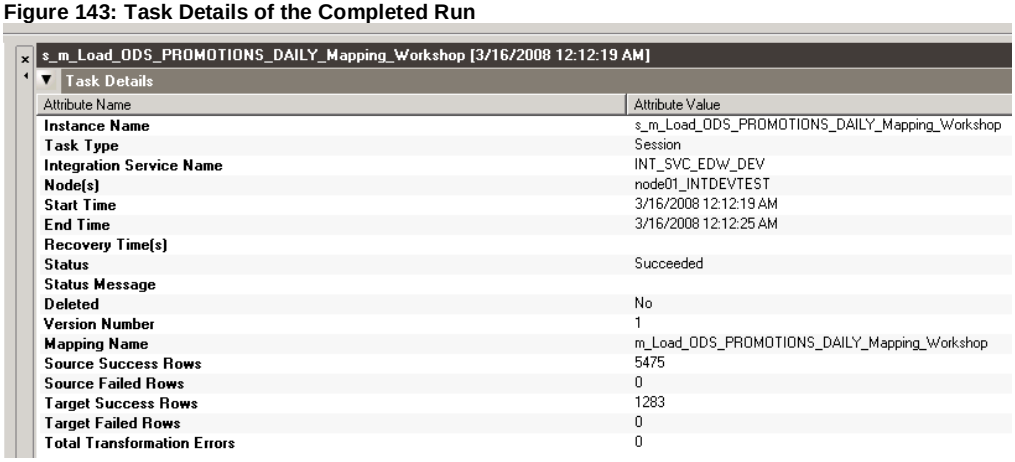
SOURCE TO TARGET FIELD MATRIX

Target Table	Target Column	Source	File	Source Column	Expression	Default Value for Nulls

Run Details

Your Task Details, Source/Target Statistics, and preview of the Target data should be similar to the figures below.

Figure 143: Task Details of the Completed Run

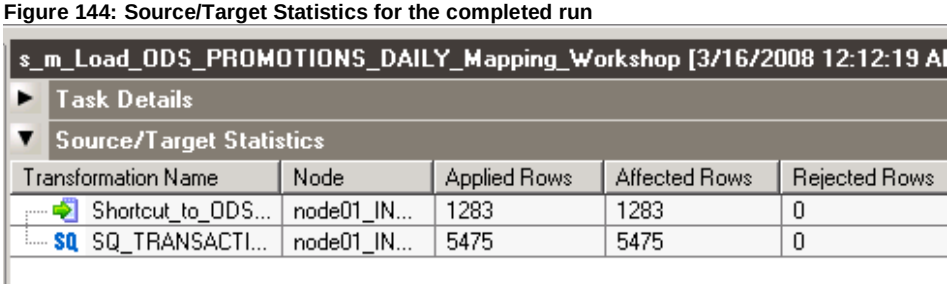


s_m_Load_ODS_PROMOTIONS_DAILY_Mapping_Workshop [3/16/2008 12:12:19 AM]

Task Details

Attribute Name	Attribute Value
Instance Name	s_m_Load_ODS_PROMOTIONS_DAILY_Mapping_Workshop
Task Type	Session
Integration Service Name	INT_SVC_EDW_DEV
Node(s)	node01_INTDEVTEST
Start Time	3/16/2008 12:12:19 AM
End Time	3/16/2008 12:12:25 AM
Recovery Time(s)	
Status	Succeeded
Status Message	
Deleted	No
Version Number	1
Mapping Name	m_Load_ODS_PROMOTIONS_DAILY_Mapping_Workshop
Source Success Rows	5475
Source Failed Rows	0
Target Success Rows	1283
Target Failed Rows	0
Total Transformation Errors	0

Figure 144: Source/Target Statistics for the completed run



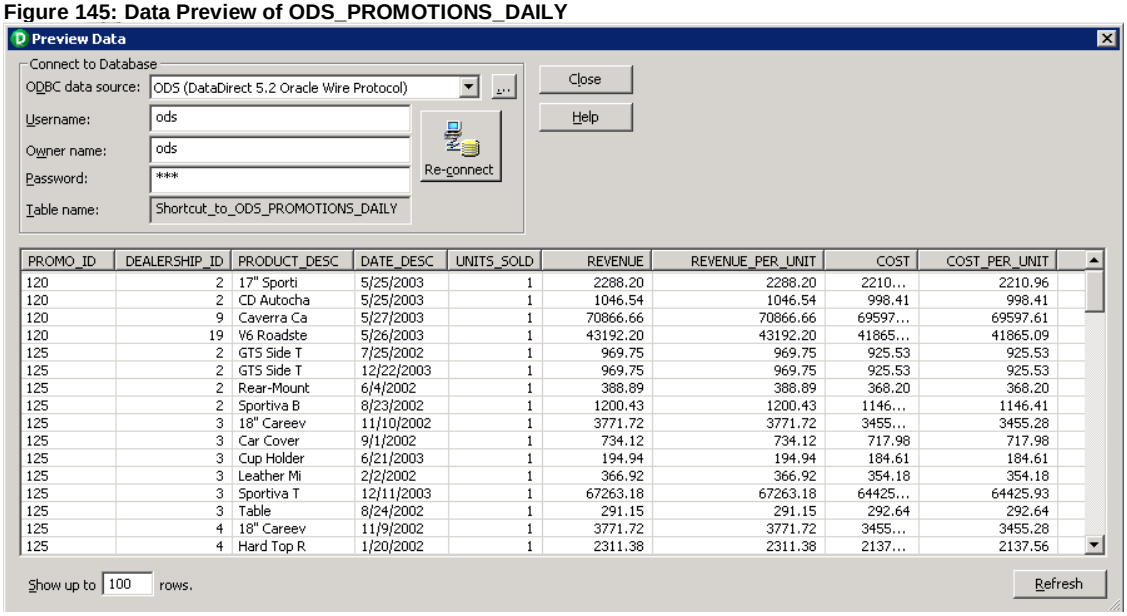
s_m_Load_ODS_PROMOTIONS_DAILY_Mapping_Workshop [3/16/2008 12:12:19 AM]

Task Details

Source/Target Statistics

Transformation Name	Node	Applied Rows	Affected Rows	Rejected Rows
Shortcut_to_ODS...	node01_IN...	1283	1283	0
SQ_TRANSACTI...	node01_IN...	5475	5475	0

Figure 145: Data Preview of ODS_PROMOTIONS_DAILY



Preview Data

Connect to Database

ODBC data source: ODS (DataDirect 5.2 Oracle Wire Protocol)

Username: ods

Owner name: ods

Password: ***

Table name: Shortcut_to_ODS_PROMOTIONS_DAILY

Re-connect

Close

Help

PROMO_ID	DEALERSHIP_ID	PRODUCT_DESC	DATE_DESC	UNITS_SOLD	REVENUE	REVENUE_PER_UNIT	COST	COST_PER_UNIT
120	2	17" Sporti	5/25/2003	1	2288.20	2288.20	2210...	2210.96
120	2	CD Autocha	5/25/2003	1	1046.54	1046.54	998.41	998.41
120	9	Caverra Ca	5/27/2003	1	70866.66	70866.66	69597...	69597.61
120	19	V6 Roadste	5/26/2003	1	43192.20	43192.20	41865...	41865.09
125	2	GTS Side T	7/25/2002	1	969.75	969.75	925.53	925.53
125	2	GTS Side T	12/22/2003	1	969.75	969.75	925.53	925.53
125	2	Rear-Mount	6/4/2002	1	388.89	388.89	368.20	368.20
125	2	Sportiva B	8/23/2002	1	1200.43	1200.43	1146...	1146.41
125	3	18" Careev	11/10/2002	1	3771.72	3771.72	3455...	3455.28
125	3	Car Cover	9/1/2002	1	734.12	734.12	717.98	717.98
125	3	Cup Holder	6/21/2003	1	194.94	194.94	184.61	184.61
125	3	Leather Mi	2/2/2002	1	366.92	366.92	354.18	354.18
125	3	Sportiva T	12/11/2003	1	67263.18	67263.18	64425...	64425.93
125	3	Table	8/24/2002	1	291.15	291.15	292.64	292.64
125	4	18" Careev	11/9/2002	1	3771.72	3771.72	3455...	3455.28
125	4	Hard Top R	1/20/2002	1	2311.38	2311.38	2137...	2137.56

Show up to 100 rows.

Refresh

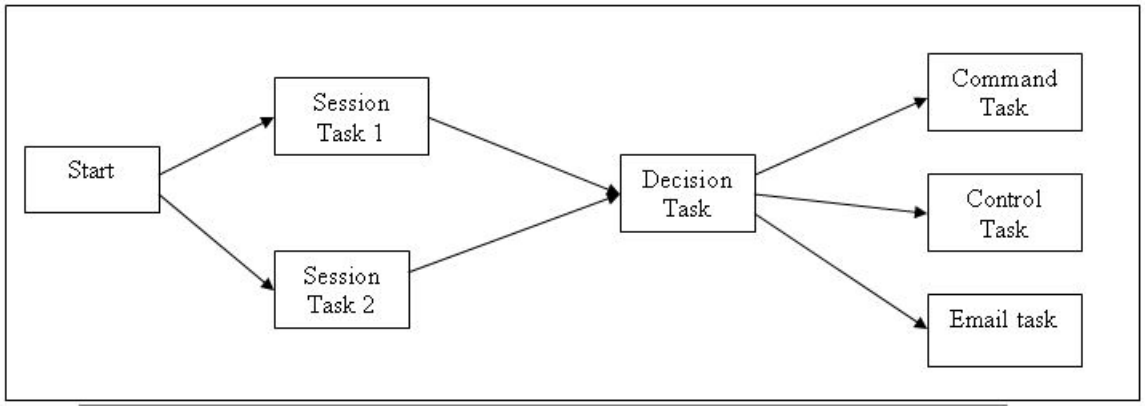
Workshop 2: Workflow Design Workshop

What to Consider

The Workflow process requires some up-front research. Before designing a Workflow, it is important to have a clear picture of the task-to-task processes.

- Design a high-level view of the Workflow and document the process within it, using a textual description to explain exactly what it is supposed to accomplish and the steps or methods it will use to accomplish its goal.

Figure 146: Example of High-Level Workflow Overview



- The load development process involves the following steps:
  - Clearly define and document all dependencies.
  - Analyze the processing resources available.
  - Develop operational requirements.
  - Develop tasks, Worklets, and Workflows based on the results.
- Create an inventory of Worklets and reusable tasks. This list is a "work in progress" list and will have to be continually updated as the project moves forward.
  - The lists are valuable to everyone, but particularly for the lead developer.
  - Making an up-front decision to make all Session, Email and Command tasks reusable will make this easier.
- The administrator or lead developer should put together a list of database connections to be used for Source and Target connection values.
- Reusable tasks must be properly documented to make it easier for other developers to determine whether they can or should use them in their own development.

- If the volume of data is sufficiently low for the available hardware to handle, you may consider volume analysis optional, developing the load process solely on the dependency analysis.
    - If the hardware is not adequate to run the Sessions concurrently, you will need to prioritize them. The highest priority within a group is usually assigned to Sessions with the most child dependencies.
- Another possible component to add into the load process is sending email. Three email options are available for notification during the load process:
    - Post-session emails can be sent after a Session completes successfully or when if fails.
    - Email tasks can be placed in Workflows before or after an event or series of events.
    - Emails can be sent when Workflows are suspended.
- Document any other information about the Workflow that is likely to be helpful in developing. Helpful information may, for example, include Source and Target database connection information; pre- or post-Workflow processing requirements; and any information about specific error handling for the Workflow.
- Create a Load Dependency analysis. This should list all Sessions by dependency, along with all other events (Informatica or other) they depend on.
    - Also, be sure to specify the dependency relationship between each Session or event, the algorithm or logic needed to test the dependency during execution, and the impact of any possible dependency test results (e.g., don't run a Session, fail a Session, fail a parent or Worklet, etc.)
- Create a Load Volume analysis. This should list all the Sources and row counts and row widths expected for each Session.
    - This should include all Lookup transformations in addition to the extract Sources. The amount of data that is read to initialize a Lookup cache can materially affect the initialization and execution time of a Session.
- The completed Workflow design should then be reviewed with one or more team members for completeness and adherence to the business requirements.
- The design document should be updated whenever the business rules change, or if more information is gathered during the build process.

Workflow Specifics

The following tips will make the Workflow development process more efficient (not in any particular order).

- When developing a sequential Workflow, use the Workflow Wizard to create Sessions in sequence. You also have the option to create dependencies between Sessions.
- Use a parameter file to define the values for parameters and variables used in a Workflow, Worklet, Mapping, or Session. A parameter file can be created using a text editor such as WordPad or Notepad. List the parameters or variables and their values in the parameter file. The use of Parameter files is covered in the Level 2 Developer course. Parameter files can contain the following types of parameters and variables:
  - Workflow variables
  - Worklet variables
  - Session parameters
  - Mapping parameters and variables
- When using parameters or variables in a Workflow, Worklet, Mapping or Session, the Integration Service checks the parameter file to determine the start value of the parameter or variable. Use a parameter file to initialize Workflow variables, Worklet variables, Mapping parameters and Mapping variables. If not defining start values for these parameters and variables, the Integration Service checks for the start value of the parameter or variable in other places.
- Session parameters **must** be defined in a parameter file. Since Session parameters do not have default values, when the Integration Service cannot locate the value of a Session parameter in the parameter file, it fails to initialize the Session.
- To include parameter or variable information for more than one Workflow, Worklet, or Session in a single parameter file, create separate sections for each object within the parameter file.
- You can create multiple parameter fields for a single Workflow, Worklet, or Session and change the file these tasks use as necessary. To specify the parameter file the Integration Service uses with a Workflow, Worklet, or Session, do either of the following:
  - Enter the parameter file name and directory in the Workflow, Worklet, or Session properties.
  - Start the Workflow, Worklet or Session using pmcmd and enter the parameter filename and directory on the command line.
- On hardware systems that are underutilized, you may be able to improve performance by processing partitioned datasets in parallel in multiple threads of the same Session instance running on the Integration Service node.
  - However, parallel execution may impair performance on over-utilized systems or systems with smaller I/O capacity.

- Incremental aggregation is useful for applying captured changes in the Source to aggregate calculations in a Session. If the Source changes only incrementally, and you can capture changes, you can configure the Session to process only those changes.
    - This allows the Integration Service to update your Target incrementally, rather than forcing it to process the entire Source and recalculate the same calculations each time you run the Session.
    - This is particularly useful for "slowly-changing dimension" tables. Slowly changing dimensions are covered in the Level 2 Developer class.
- Target Load Based Strategies:
    - Loading directly into the Target is possible when the data is going to be bulk loaded.
    - Load into flat files and bulk load using an external loader.
    - Load into a mirror database.
- From the Workflow Manager Tools menu, select Options and deselect the option to "Show full names of task." This will show the entire name of all tasks in the Workflow.

Workshop

Scenario:

- All the staging tables need to be loaded in a single Workflow.

Goals:

- Design and create a Workflow to load all of the staging tables.

Duration:

120 minutes

Instructions

Mappings Required

This section contains a listing of the Mappings that will be used in the workflow:

- m_Load_STG_PAYMENT_TYPE
- m_Load_STG_Product
- m_Load_STG_Dealership
- m_Load_STG_PROMOTIONS
- m_Load_STG_CUSTOMERS
- m_Load_STG_TRANSACTIONS
- m_Load_STG_EMPLOYEES

For your convenience, reusable Sessions have been created for these mappings. You can **COPY** them from the SC_DATA_STRUCTURES folder to your folder. (One or more of these Sessions may already be in your Sessions subfolder.) Remember to use the Repository Manager to copy the sessions. If the copy wizard asks to resolve any conflicts, tell it to replace old definitions with new ones.

The names of the sessions are:

- s_m_Load_STG_PAYMENT_TYPE
- s_m_Load_STG_PRODUCT
- s_m_Load_STG_DEALERSHIP
- s_m_Load_STG_PROMOTIONS
- s_m_Load_STG_CUSTOMERS
- s_m_Load_STG_TRANSACTIONS
- s_m_Load_STG_EMPLOYEES

Workflow Details

1. Name the workflow **wkf_LOAD_ALL_STAGING_TABLES_*xx***.
   The Workflow needs to start at a certain time each day. For this workshop, set the start time to a couple of minutes after you complete the Workflow. Remember that the start time is relative to the time on the Integration Service process machine, not a developer's local machine.
2. No Session can begin until an indicator file shows up. The indicator file will be named fileind*xx*.txt, and will be created by you using a text editor. You will need to place this file in the directory indicated by the Instructor after you start the Workflow. (If you are in a UNIX environment, you can skip this requirement.)
3. In order to utilize the CPU in a more efficient manner, you will want to run some of the Sessions sequentially and some of them concurrently.
   a. The Sessions s_m_Load_STG_PAYMENT_TYPE, s_m_Load_STG_PRODUCT, s_m_STG_DEALERSHIP and s_m_Load_STG_PROMOTIONS can all be run sequentially.
   b. The Session s_m_Load_STG_CUSTOMERS can be run concurrently with the Sessions in point a.
   c. If any of these sessions fail, an email should be sent to the administrator, and the Workflow aborted. Use admin@anycompany.com as the email user name.
   d. The session containing Mapping m_STG_EMPLOYEES should be run only after the five previously-mentioned Sessions complete successfully.
   e. The Session s_m_Load_STG_TRANSACTIONS can be run concurrently with the session s_m_STG_EMPLOYEES.
   f. If either of these Sessions fails, an email should be sent to the Administrator.
4. All Sessions truncate the Target tables and should be pointed to the correct relational database connections.
5. Management wants the Workflow to run for a maximum of 50 minutes. If the Workflow takes longer than 50 minutes, an email must be sent to the administrator. If the Workflow completes on time, the email should not be sent.

   This is a subtle point. It means you must stop the workflow nicely before the timer task has a chance to finish running and the "long running workflow" email gets sent. Consider which task to use and where to place it in the workflow so that when all other tasks run to completion properly, the "long running workflow" email is not sent.

Final Point

More than one solution is possible. You will know that your solution has worked when all the Sessions complete successfully.