Mapping Analyst for Excel Guide

Informatica® PowerCenter®
(Version 8.6.1)
# Table of Contents

**Preface** ................................................................. vii  
Informatica Resources ................................................ vii  
  Informatica Customer Portal ..................................... vii  
  Informatica Documentation ....................................... vii  
  Informatica Web Site ................................................ vii  
  Informatica How-To Library ...................................... vii  
  Informatica Knowledge Base ..................................... viii  
  Informatica Global Customer Support ........................... viii  

**Chapter 1: Understanding Mapping Analyst for Excel**  .................. 1  
Overview ........................................................................ 1  
Architecture .................................................................... 1  
  Mapping Specifications ................................................ 1  
  Metamaps .................................................................... 2  
  Mapping Analyst for Excel Process ............................... 2  
Mapping Specification Templates ...................................... 3  
  Mapping Template ........................................................ 3  
  Source-Target-Matrix Template .................................... 3  

**Chapter 2: Configuring One Mapping** ....................................... 5  
Overview ........................................................................ 5  
Copying and Renaming the Mapping Template .................... 6  
Main Worksheet ............................................................. 6  
Source and Target Worksheets .......................................... 7  
  Database/Schema Name Section .................................... 7  
  Classes/Entities/Tables/Records Section ......................... 7  
  Attributes/Columns/Fields Section .............................. 7  
  Types Section ............................................................. 8  
  Mapping Analyst for Excel Datatypes ............................ 8  
  Configuring Target Ports ............................................. 8  
Functions ........................................................................ 9  
Transformations ................................................................ 9  
  Joiner Transformation ................................................ 9  
  Lookup Transformation ............................................... 10  
  Filter Transformation .................................................. 11  
  Aggregator Transformation .......................................... 11  
  Expression Transformation .......................................... 12  
Validating the Mapping Specification ................................ 12  

**Chapter 3: Configuring One Mapping - Example** ......................... 15  
Overview ........................................................................ 15  
Step 1. Establish the PowerCenter Environment .................. 16
Index ................................................................. 67
Preface

The PowerCenter Mapping Analyst for Excel Guide is written for business analysts who want to use Microsoft Excel to develop PowerCenter mappings. It is also written for PowerCenter developers who want to use the Microsoft Excel files to begin developing PowerCenter mappings. This book assumes you have a working knowledge of Microsoft Excel.

Informatica Resources

Informatica Customer Portal

As an Informatica customer, you can access the Informatica Customer Portal site at http://my.informatica.com. The site contains product information, user group information, newsletters, access to the Informatica customer support case management system (ATLAS), the Informatica How-To Library, the Informatica Knowledge Base, Informatica Documentation Center, and access to the Informatica user community.

Informatica Documentation

The Informatica Documentation team takes every effort to create accurate, usable documentation. If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com. We will use your feedback to improve our documentation. Let us know if we can contact you regarding your comments.

The Documentation team updates documentation as needed. To get the latest documentation for your product, navigate to the Informatica Documentation Center from http://my.informatica.com.

Informatica Web Site

You can access the Informatica corporate web site at http://www.informatica.com. The site contains information about Informatica, its background, upcoming events, and sales offices. You will also find product and partner information. The services area of the site includes important information about technical support, training and education, and implementation services.

Informatica How-To Library

As an Informatica customer, you can access the Informatica How-To Library at http://my.informatica.com. The How-To Library is a collection of resources to help you learn more about Informatica products and features. It includes articles and interactive demonstrations that provide solutions to common problems, compare features and behaviors, and guide you through performing specific real-world tasks.
Informatica Knowledge Base

As an Informatica customer, you can access the Informatica Knowledge Base at http://my.informatica.com. Use the Knowledge Base to search for documented solutions to known technical issues about Informatica products. You can also find answers to frequently asked questions, technical white papers, and technical tips.

Informatica Global Customer Support

There are many ways to access Informatica Global Customer Support. You can contact a Customer Support Center through telephone, email, or the WebSupport Service.

Use the following email addresses to contact Informatica Global Customer Support:
- support@informatica.com for technical inquiries
- support_admin@informatica.com for general customer service requests

WebSupport requires a user name and password. You can request a user name and password at http://my.informatica.com.

Use the following telephone numbers to contact Informatica Global Customer Support:

<table>
<thead>
<tr>
<th>North America / South America</th>
<th>Europe / Middle East / Africa</th>
<th>Asia / Australia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informatica Corporation</td>
<td>Informatica Software Ltd.</td>
<td>Informatica Business Solutions Pvt. Ltd.</td>
</tr>
<tr>
<td>Headquarters</td>
<td>6 Waltham Park</td>
<td>Diamond District</td>
</tr>
<tr>
<td>100 Cardinal Way</td>
<td>Waltham Road, White Waltham</td>
<td>Tower B, 3rd Floor</td>
</tr>
<tr>
<td>Redwood City, California</td>
<td>Maidenhead, Berkshire</td>
<td>150 Airport Road</td>
</tr>
<tr>
<td>94063</td>
<td>SL6 3TN</td>
<td>Bangalore 560008</td>
</tr>
<tr>
<td>United States</td>
<td>United Kingdom</td>
<td>India</td>
</tr>
<tr>
<td>Toll Free</td>
<td>Toll Free</td>
<td>Toll Free</td>
</tr>
<tr>
<td>+1 877 463 2435</td>
<td>00 800 4632 4357</td>
<td>Australia: 1 800 151 830</td>
</tr>
<tr>
<td>Standard Rate</td>
<td>Standard Rate</td>
<td>Singapore: 001 800 4632</td>
</tr>
<tr>
<td>Brazil: +55 11 3523 7761</td>
<td>Belgium: +32 15 281 702</td>
<td>4357</td>
</tr>
<tr>
<td>Mexico: +52 55 1168 9763</td>
<td>France: +33 1 41 38 92 26</td>
<td>Standard Rate</td>
</tr>
<tr>
<td>United States: +1 650 385 5800</td>
<td>Germany: +49 1805 702 702</td>
<td>India: +91 80 4112 5738</td>
</tr>
<tr>
<td></td>
<td>Netherlands: +31 306 022 797</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Spain and Portugal: +34 93 480 3760</td>
<td></td>
</tr>
<tr>
<td></td>
<td>United Kingdom: +44 1628 511 445</td>
<td></td>
</tr>
</tbody>
</table>
Overview

Use Mapping Analyst for Excel to configure PowerCenter mappings in Microsoft Office Excel. The following individuals can collaborate when creating PowerCenter mappings:

- **Business analyst.** Creates a mapping specification in Microsoft Office Excel to define a mapping that can include sources, targets, and transformations. A business analyst is familiar with project requirements and source and target data, but is not a PowerCenter user.

- **PowerCenter developer.** Imports the mapping specification using the Repository Manager to create the corresponding PowerCenter objects. The PowerCenter developer can edit the objects, implement additional functionality that might be required, and run a workflow generated from the mapping. The PowerCenter developer can use the Repository Manager to export any valid mapping to a mapping specification for a business analyst to review in Microsoft Excel.

For example, a business analyst wants to merge employee data from multiple sources into a single data warehouse but is not familiar enough with the PowerCenter Client to create the required mapping. The business analyst can use the templates included with the PowerCenter Client to create a mapping specification that describes the required sources, transformations, and targets. A PowerCenter developer can then import the mapping specification into a PowerCenter repository, continue development of the mapping in the PowerCenter Client, and export the completed mapping for review by the business analyst.

Architecture

Mapping Analyst for Excel includes mapping specifications and metamaps.

Mapping Specifications

A mapping specification is a Microsoft Excel file that includes metadata to import into the PowerCenter repository. Use a mapping specification to define sources or targets or to define a mapping by describing the
sources and targets and the transformations between sources and targets. Mapping Analyst for Excel includes templates for mapping specifications. Use the mapping specification templates as a starting point to develop mapping specifications in Excel. You can also create a custom mapping specification template.

When you create a mapping specification, you do not have to define all objects. For example, if you do not define transformations, the Repository Service imports only sources and targets.

**Metamaps**

A metamap is a hierarchical representation of the metadata in a mapping specification. A metamap determines how the Repository Service reads the metadata. Each mapping specification has an associated metamap. You must use the same metamap for each mapping specification you create from a template.

The metamap contains multiple worksheets that define import and export options and the cell location of metadata in a mapping specification.

Mapping Analyst for Excel includes metamaps in XML and Excel formats. Use the XML version of the metamap when you import mapping specifications or export mappings. Use the Excel version of the metamap to modify metamap properties when you create a custom mapping specification template.

**Note:** If you create a mapping specification from a template included with Mapping Analyst for Excel, you do not need to modify the metamap.

**Mapping Analyst for Excel Process**

The following figure describes the Mapping Analyst for Excel process:

To work with Mapping Analyst for Excel, use the following process:

1. Create a mapping specification based on a Mapping Analyst for Excel template or on a custom mapping specification template.
2. If you create a custom mapping specification template, modify the associated metamap to read the custom template.
3. Use the Repository Manager to import the mapping specification. The Repository Service creates the PowerCenter objects.
4. Complete the PowerCenter mapping in the PowerCenter Designer.
5. Use the Repository Manager to export PowerCenter metadata to a mapping specification for review by a business analyst.
Mapping Specification Templates

Mapping Analyst for Excel includes the following mapping specification templates and metamaps:

<table>
<thead>
<tr>
<th>Template Name</th>
<th>Template File</th>
<th>Associated Metamap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping template (with validation)</td>
<td>MappingTemplate.xls</td>
<td>MappingTemplate-MetaMap.xls</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MappingTemplate-MetaMap.xml</td>
</tr>
<tr>
<td>Mapping template (without validation)</td>
<td>MappingTemplate_NoValidation.xls</td>
<td>MappingTemplate-MetaMap.xls</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MappingTemplate-MetaMap.xml</td>
</tr>
<tr>
<td>Source-Target-Matrix template</td>
<td>source-target-matrix.xls</td>
<td>source-target-matrix-metamap.xls</td>
</tr>
<tr>
<td></td>
<td></td>
<td>source-target-matrix-metamap.xml</td>
</tr>
</tbody>
</table>

The mapping specification templates and metamaps are located in the following directory:

`<PowerCenter_Client_installation_directory>\bin\Sample Specifications`

Mapping Template

The Mapping template contains separate worksheets for overview information, sources, targets, and functions. Use this template to create a mapping specification for a single mapping. Mapping specifications based on the Mapping template can contain sources and targets and Joiner, Filter, Expression, Lookup, and Aggregator transformations.

The Mapping template includes a template with validation and a template without validation. Use the template with validation to validate metadata values in the mapping specification. Use the template without validation if you want to avoid enabling macros on your machine.

Source-Target-Matrix Template

The Source-Target-Matrix template includes a separate worksheet for each mapping in the mapping specification. Use this template to create multiple mappings in a single mapping specification. Mapping specifications based on the Source-Target-Matrix template can contain sources, targets, and Expression transformations.

The Source-Target-Matrix repeating sheet file, `source-target-matrix-mirrepeatingssheet.xls`, defines the common structure for multiple mappings defined in a mapping specification that is based on the Source-Target-Matrix template. The metamap for the Source-Target-Matrix template reads from the repeating sheet file to define the cell location of metadata in the template.
Overview

A business analyst uses the Mapping template to create a mapping specification for a single mapping. The Mapping template contains separate worksheets for overview information, sources, targets, and functions. Mapping specifications based on the Mapping template can contain sources and targets and Joiner, Filter, Expression, Lookup, and Aggregator transformations. A PowerCenter developer imports the mapping specification using the Repository Manager to create the corresponding PowerCenter objects. During import, the Repository Service names each mapping “Mapping.” The PowerCenter developer can change the mapping name in the Designer.

Before you create a mapping specification based on the Mapping template, choose whether to use a template with or without validation. The structure of the template without validation is the same as the template with validation.

To create a mapping specification based on the Mapping template, a business analyst completes the following steps:

1. Copy and rename the template.
2. Configure the Main worksheet.
3. Configure sources and targets.
4. Optionally add functions to the Functions worksheet.
5. Configure transformations.
6. Based on the mapping specification template type, click Validate to validate the transformations.
Copying and Renaming the Mapping Template

To create a mapping specification based on the Informatica Mapping template, copy and rename the template. You can find the Mapping template, MappingTemplate.xls, in the following directory:

<PowerCenter Client installation directory>\bin\Sample Specifications

Main Worksheet

Use the Main worksheet to enter general information for the mapping, such as the name of the source or target database or file system. The Main worksheet includes the following sections:

♦ Model. Entity that contains all mapping metadata.
♦ Data Store. Contains the source and target database name and type.

In general, when you configure a mapping specification, you do not need to enter all requested metadata. However, certain metadata is required to import the mapping specification into the repository.

Enter the following properties in the Model section:

<table>
<thead>
<tr>
<th>Property</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Name</td>
<td>Required</td>
<td>Name of the model. Mapping Analyst for Excel requires a model name during the import process. However, the Repository Service does not import this value.</td>
</tr>
<tr>
<td>Model Description</td>
<td>Optional</td>
<td>Description for the model. The Repository Service does not import this value.</td>
</tr>
<tr>
<td>Model Creation Time</td>
<td>Optional</td>
<td>Time the model was created. The Repository Service does not import this value.</td>
</tr>
</tbody>
</table>

Enter the following properties in the Source Data Store and Target Data Store sections:

<table>
<thead>
<tr>
<th>Property</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Required</td>
<td>Name of the source or target data store. Source and target data stores must have different names. During import, the Repository Service concatenates the Source Data Store Name and the Source Schema Name using the format &lt;SourceDataStoreName&gt;_&lt;SourceSchemaName&gt;. It assigns the concatenated name to the database name for the source definition.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional</td>
<td>Description for the source or target data store. The Repository Service does not import this value.</td>
</tr>
<tr>
<td>System Type</td>
<td>Required</td>
<td>Source or target database or file system type. The Repository Service assigns this database type to the source or target definition.</td>
</tr>
<tr>
<td>Schema Names</td>
<td>Required</td>
<td>Name of the source or target database or file system. During import, the Repository Service concatenates the Source Data Store Name and the Source Schema Name using the format &lt;SourceDataStoreName&gt;_&lt;SourceSchemaName&gt;. It assigns the concatenated name to the database name for the source definition.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional</td>
<td>Description for the source or target database or file system. The Repository Service does not import this value.</td>
</tr>
</tbody>
</table>
Source and Target Worksheets

Use the Source and Target worksheets to configure the sources and targets for the mapping. You can also configure transformations. The Source and Target worksheets include the following sections:

- **Database/Schema Name.** Name of the source or target database or file system.
- **Classes/Entities/Tables/Records.** Source or target table and file information.
- **Attributes/Columns/Fields.** Source or target column and field details.
- **Types.** Datatype for each column or field.
- **Transformations.** Transformations to include in the mapping. The worksheet you configure determines the type of transformations you can configure. For more information, see “Transformations” on page 9.

**Database/Schema Name Section**

In the Database/Schema name column, optionally enter the name of the source or target database or file system. This property must be blank or must match the value for the Schema Name property on the Main worksheet.

**Classes/Entities/Tables/Records Section**

The Classes/Entities/Tables/Records section includes source or target table and file information.

Enter the following information for each source and target in the mapping:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical Name</td>
<td>Required</td>
<td>Table or file name. The Repository Service creates a PowerCenter source or target definition with this name.</td>
</tr>
<tr>
<td>Business Name</td>
<td>Optional</td>
<td>Business name for the table or file name. The Repository Service assigns this value to the business name of the source or target definition. If you do not enter a value, the Repository Service uses the physical name for the business name.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional</td>
<td>Description for the table or file name. The Repository Service assigns this value to the description of the source or target definition.</td>
</tr>
</tbody>
</table>

**Attributes/Columns/Fields Section**

The Attributes/Columns/Fields section contains source or target column and field details.

Enter the following information for each source and target in the mapping:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical Name</td>
<td>Required</td>
<td>Column name. The Repository Service creates a port in the source or target definition with this name.</td>
</tr>
<tr>
<td>Business Name</td>
<td>Optional</td>
<td>Business name for the column. The Repository Service assigns this value to the port business name. If you do not enter a value, the Repository Service uses the physical name for the business name.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional</td>
<td>Description for the column. The Repository Service assigns this value to the port description.</td>
</tr>
</tbody>
</table>
Types Section

The Types section contains the datatype for each column or field.

Enter the following information for each source and target in the mapping:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datatype</td>
<td>Required</td>
<td>Datatype of the column. For sources and targets in relational databases, enter the Mapping Analyst for Excel datatype. The Repository Service assigns the corresponding native datatype to the port in the source or target definition. For more information, see &quot;Mapping Analyst for Excel Datatypes&quot; on page 8.</td>
</tr>
<tr>
<td>Length</td>
<td>Required</td>
<td>Length of the column. The Repository Service assigns this value to the precision of the port.</td>
</tr>
<tr>
<td>Scale</td>
<td>Required</td>
<td>Scale of the column. The Repository Service assigns this value to the scale of the port.</td>
</tr>
</tbody>
</table>

Mapping Analyst for Excel Datatypes

When you import a mapping specification, the Repository Service uses the associated metamap to define mapping datatypes. The metamap lists native datatypes for each supported system or file type and the corresponding Mapping Analyst for Excel datatype.

To ensure the accuracy of source and target datatypes for relational databases after you import a mapping specification, use the Mapping Analyst for Excel datatypes when you configure relational sources and targets. For example, to create an Integer column for a Microsoft SQL Server table, enter SQL_INTEGER for the datatype in the mapping specification.

The metamap lists source and target datatypes and the corresponding Mapping Analyst for Excel datatypes. You can locate the list on the MITI_Datatypes worksheet. Mapping Analyst for Excel datatypes appear in the MIR Datatype column.

Tip: If you do not have exact or complete datatype information, you can use the description column in the mapping specification to provide information for the PowerCenter developer.

Configuring Target Ports

To configure a target port, enter an expression in the Formula/Expression column for the target port. The expression can be the name of a source or lookup table port or an aggregate or non-aggregate expression. Unconfigured target ports appear in a PowerCenter target definition but are not connected to the PowerCenter mapping pipeline.

To avoid validation errors when defining target ports, use the source table name and column name as follows:

\(<\text{TableName}.<\text{portname}>\)

Table and port names are case sensitive.

The following table describes how target port configuration corresponds to a PowerCenter mapping:

<table>
<thead>
<tr>
<th>Formula/Expression</th>
<th>Example</th>
<th>PowerCenter Mapping Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of a source port</td>
<td>INVENTORY.ItemName</td>
<td>Connects the pipeline related to the source port to the target port.</td>
</tr>
<tr>
<td>Name of a lookup table port</td>
<td>REVENUE.MonthlyRecognizedRevenue</td>
<td>Includes return values from the lookup table that the Integration Service returns when the lookup condition is met. Use when the mapping pipeline includes a Lookup transformation. For more information, see &quot;Lookup Transformation&quot; on page 10.</td>
</tr>
</tbody>
</table>
Functions

The Functions worksheet lists the PowerCenter functions that you can use to configure transformation expressions. Functions that you can use in aggregate expressions are labeled Aggregation. You can add custom functions to the Functions worksheet.

The validation macros compare functions used in expressions with the list on the Functions worksheet.

To add functions:

1. On the Functions worksheet, add the custom function name to the list.
2. If the function is an aggregate function, label the function Aggregation in the Type column.

Transformations

You add transformations to a mapping specification on the following worksheets:

- **Source worksheet.** Configure Joiner, Lookup, and Filter transformations.
- **Target worksheet.** Configure Expression and Aggregator transformations.

Joiner Transformation

You can join source data from two related heterogeneous sources residing in different locations or file systems. You join source data by matching one or more pairs of columns between the two sources. You configure a join on the Source worksheet of the mapping specification. A join in the mapping specification becomes a Joiner transformation in a PowerCenter mapping.

On the Source worksheet, enter the join information for the appropriate port in the left or master table. For example, if you are joining two sources by a column named StoreID, enter the join information in the row that describes the StoreID port of the left or master table.

The following table describes the information you configure for a Joiner transformation:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right Table</td>
<td>Name of the right or detail table to join with the left or master table. The table must be configured on the Source worksheet.</td>
</tr>
</tbody>
</table>
The following figure shows a left or master table named SALES joined with a right or detail table named REVENUE by the StoreID column with an Outer join:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right Column</td>
<td>Column name in the right or detail table to match with the column in the left or master table. The Repository Service uses these columns to define the join condition.</td>
</tr>
</tbody>
</table>
| Type | Type of join for the Joiner transformation. Enter one of the following types:  
- LEFT. Left outer join. The Repository Service creates a Joiner transformation with a Master Outer join.  
- RIGHT. Right outer join. The Repository Service creates a Joiner transformation with a Detail Outer join.  
- INNER. Inner join. The Repository Service creates a Joiner transformation with a Normal join.  
- OUTER. Outer join. The Repository Service creates a Joiner transformation with a Full Outer join. |

### Lookup Transformation

You can configure a lookup to find data outside of the mapping pipeline. You configure a lookup on the Source worksheet of the mapping specification. A lookup in the mapping specification becomes a connected Lookup transformation in a PowerCenter mapping.

You can perform a lookup on any source defined in the mapping specification. To perform a lookup on a table that is not used as a mapping source, you can include the lookup table in the file as a source table without connecting it to mapping targets.

To create a lookup, define the following information:
- Lookup condition
- Return values

#### Lookup Condition

The Integration Service finds data in the lookup source with a lookup condition. The lookup condition is similar to the WHERE clause in an SQL query.

Define the condition for the appropriate column in the lookup table. For example, to return values from a REVENUE lookup table for matching store IDs, enter the following lookup condition in the Lookups column for the row that describes the StoreID port of the REVENUE table:

```sql
REVENUE.StoreID=SALES.StoreID
```

#### Return Values

Return values are the values from the lookup table that the Integration Service returns when the lookup condition is met.
Define return values by entering the name of a lookup table column as an expression for a target column on the Target worksheet. For example, to include monthly revenue data from the lookup table in the mapping, define REVENUE.MonthlyRecognizedRevenue as the expression for the MonthlyRecognizedRevenue target column on the Target worksheet.

Filter Transformation

You can configure a filter to remove source data from the mapping pipeline. You configure a filter on the Source worksheet of the mapping specification. A filter defined in the mapping specification becomes a Filter transformation in a PowerCenter mapping.

Configure a filter by defining a filter condition. The filter condition is an expression that returns TRUE or FALSE. For example, use the following expression to filter out transactions with negative values, such as returns:

\[ \text{TransactionAmount} > 0 \]

You can also specify multiple components for a condition using the AND and OR logical operators.

Define a filter condition for the appropriate port in the source table. For example, enter the TransactionAmount condition in the Filters column for the row that describes the TransactionAmount port.

Aggregator Transformation

You can configure an aggregate expression to perform calculations on multiple values in a port. You configure an aggregate expression on the Target worksheet. An aggregate expression defined in the mapping specification becomes an Aggregator transformation in a PowerCenter mapping.

Use aggregate functions to perform calculations on multiple values in a port. For example, the following aggregate expression evaluates all values in the Quantity port of the SALES table and returns the highest value:

\[ \text{MAX(SALES.Quantity)} \]

You can use an aggregate function, conditional clauses, and non-aggregate functions in an aggregate expression. You can also nest one aggregate function within another aggregate function, for example:

\[ \text{MAX( COUNT( SALES.Quantity))} \]

In the Mapping template, aggregate functions are marked “Aggregation” in the list of functions on the Functions worksheet.

Define an aggregate expression in the Formula/Expression column for the row where you want return values to be written.

When you validate the Target worksheet in the mapping specification, Microsoft Excel defines the expression type in the Transformations Type column.

The following figure shows an aggregate expression configured for the StoreSales port of the T_SALES target table:
Grouping Values

In PowerCenter, you use group by ports in the Aggregator transformation to group values for aggregate calculations. The result of an aggregate expression varies depending on the group by ports you configure.

For example, when the Integration Service calculates the following aggregate expression with no group by ports defined, it finds the total sales from all transactions:

```
SUM( SALES.TransactionAmount )
```

However, if you use the same expression, and you group by the StoreID port, the Integration Service returns the total sales for each store ID.

When a PowerCenter developer imports a mapping specification, the Repository Service creates a single Aggregator transformation for all aggregate expressions defined for the target. Any port in the Aggregator transformation without an aggregate expression becomes a group by port.

Use the Transformations Description column to indicate which group by ports you want to use. After importing the mapping specification, the PowerCenter developer can view that information in the Aggregator transformation and configure group by ports appropriately.

Expression Transformation

You can configure an expression to calculate values in a single row for each row in a port. You configure an expression on the Target worksheet of the mapping specification. An expression defined in the mapping specification becomes an Expression transformation in a PowerCenter mapping.

Use an expression to calculate values in a single row for each row in a port. For example, the following expression increases the cost of each item by five percent:

```
INVENTORY.Cost + (INVENTORY.Cost * .05)
```

You can use conditional clauses and non-aggregate functions in non-aggregate expressions. You can also nest non-aggregate functions.

When you validate the Target worksheet in the mapping specification, Microsoft Excel defines the expression type in the Transformations Type column.

Validating the Mapping Specification

The Mapping template with validation includes macros to perform validation of the Source and Target worksheets. The macros perform the following validation:

- Expressions contain columns that are defined on the Source or Target worksheet.
- Functions used in the mapping are listed on the Functions worksheet.
- Function types are defined in the Type column in the Target worksheet based on the definitions on the Functions worksheet.

To perform validation:

1. On the Source worksheet, click Validate.
   The validation notifies you of any error.
2. Correct errors and click Validate again.
3. On the Target worksheet, click Validate.
   The validation fills in the transformation types, “expr” for expression and “aggr” for aggregation and notifies you of any error.
4. Correct errors and click Validate again.
5. Save the changes.
Overview

This example uses the Mapping template with validation to configure a mapping that aggregates sales and revenue data into a data warehouse for analysis. In addition, the example mapping performs the following tasks:

♦ Uses the store ID to look up monthly revenue values in a source table.
♦ Filters negative transactions, such as returns, out of the pipeline.
♦ Calculates the total sales for each store.

The mapping reads from the following source tables located in an Oracle schema named Sales:

♦ Sales. Stores the current month’s sales data for each store.
♦ Revenue. Stores the revenue recognized in the current month from sales that have been occurring at each store.

The mapping writes the aggregated data to a target table named T_Sales located in an Oracle schema named DM.

To create this example mapping, the business analyst and PowerCenter developer complete the following steps:

1. Establish the PowerCenter environment (Developer). The PowerCenter developer configures the database connectivity, designates a repository folder to store the imported objects, and creates the source and target tables.
2. Configure the mapping specification in Microsoft Excel (Business Analyst). The business analyst creates a mapping specification from the Mapping template and configures the Main, Source, and Target worksheets.
3. Validate the mapping specification. (Business Analyst).
4. Save the mapping specification to a shared location (Business Analyst).
5. Import the mapping specification (Developer).
6. Edit the PowerCenter mapping and generate a workflow (Developer).

Step 1. Establish the PowerCenter Environment

Before the business analyst creates a mapping specification, the PowerCenter developer completes the following tasks to establish the PowerCenter environment:

- **Configure database connections.** Use the Workflow Manager to configure connection objects so that the Integration Service can connect to the source and target databases.
- **Designate a repository folder to store the imported objects.** Use the Repository Manager to select an existing folder or to create a folder for the PowerCenter objects that are imported from the mapping specification.
- **Create source and target tables.** Use Oracle SQL Plus to run SQL statements to use the sample data in this example.

To use the source tables included in this example, use Oracle SQL Plus to run the following SQL statements to create the Sales and Revenue source tables in an Oracle schema named Sales:

```sql
Connect sales/sales
CREATE TABLE sales.sales (StoreID NUMBER, OldTransactionNumber NUMBER, TransactionDate DATE, TransactionAmount NUMBER);

Insert into sales.sales values (1,100,'1-JAN-2008',1000);
Insert into sales.sales values (1,101,'1-JAN-2008',2000);
Insert into sales.sales values (1,102,'3-JAN-2008',3000);
Insert into sales.sales values (1,103,'3-JAN-2008',4000);
Insert into sales.sales values (2,104,'2-JAN-2008',1000);
Insert into sales.sales values (2,105,'2-JAN-2008',2000);
Insert into sales.sales values (2,106,'4-JAN-2008',2000);
Insert into sales.sales values (2,107,'4-JAN-2008',3000);

CREATE TABLE sales.revenue (StoreID NUMBER, MonthlyRecognizedRevenue NUMBER);

Insert into sales.revenue values (1,8000);
Insert into sales.revenue values (2,7000);
Commit;
```

To use the target table included in this example, use Oracle SQL Plus to run the following SQL statements to create the T_Sales target table in an Oracle schema named DM:

```sql
Connect DM/DM
CREATE TABLE DM.T_Sales (StoreID NUMBER, TransactionDate DATE, StoreSales NUMBER, MonthlyRecognizedRevenue NUMBER);
```

Step 2. Configure the Mapping Specification in Microsoft Excel

The business analyst completes the following tasks to configure the mapping specification in Microsoft Excel:

- Create a mapping specification based on the Mapping template.
- Configure the Main worksheet.
- Configure the Source worksheet.
- Configure the Target worksheet.

The Functions worksheet lists a set of functions that you can use to configure transformation expressions. In this example, you do not need to configure the Functions worksheet.

Create a Mapping Specification

To create a mapping specification based on the Informatica Mapping template, copy and rename the template. You can find the Mapping template, MappingTemplate.xls, in the following directory:

<PowerCenter Client installation directory>\bin\Sample Specifications

For example, copy and rename the template AggregateMapping.xls.

Configure the Main Worksheet

The Main worksheet defines metadata about the mapping specification, mapping, and source and target data sources.

In general, when you configure a mapping specification, you do not need to enter all requested metadata. However, certain metadata is required to import the mapping specification into the repository.

The following table describes the required metadata on the Main worksheet:

<table>
<thead>
<tr>
<th>Information</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Name</td>
<td>Name of the model that contains all of the mapping metadata.</td>
</tr>
<tr>
<td>Source Data Store Name</td>
<td>Name of the source data store that contains the mapping sources.</td>
</tr>
<tr>
<td>Source System Type</td>
<td>Type of source database. For example, enter Oracle.</td>
</tr>
<tr>
<td>Source Schema Names</td>
<td>Name of the source schema. For example, enter Sales.</td>
</tr>
<tr>
<td>Target Data Store Name</td>
<td>Name of the target data store that contains the mapping targets. Cannot be identical to the Source Data Store Name.</td>
</tr>
<tr>
<td>Target System Type</td>
<td>Type of target database. For example, enter Oracle.</td>
</tr>
<tr>
<td>Target Schema name</td>
<td>Name of the target schema. For example, enter DM.</td>
</tr>
</tbody>
</table>

Optionally, you can enter a model description and creation time as well as descriptions for the source, source schemas, target, and target schemas.

Configure the Source Worksheet

Use the Source worksheet to define the sources. You can also define join, lookup, and filter metadata for the mapping. In this example, you configure the following metadata:

- **Source tables named SALES and REVENUE.** A source defined in the mapping specification is imported as a source definition and Source Qualifier transformation in a PowerCenter mapping.

- **Lookup condition.** You can configure a lookup condition to find data in a source defined in the mapping specification. A lookup in the mapping specification becomes a connected Lookup transformation in a PowerCenter mapping. Define the lookup condition in the appropriate column in the source table. This example uses the following lookup condition to return values from the REVENUE source table for matching store IDs:

  REVENUE.StoreID=SALES.StoreID

- **Filter condition.** You can configure a filter to remove source data from the mapping pipeline. A filter defined in the mapping specification becomes a Filter transformation in a PowerCenter mapping. Define the filter condition in the appropriate column in the source table. This example uses the following filter condition to remove transactions with negative values, such as returns:

  TransactionAmount > 0
To configure the Source worksheet:

1. On the Source worksheet, enter the following information:

<table>
<thead>
<tr>
<th>Classes/Entities/Tables/Records Physical Name</th>
<th>Attributes/Columns/Fields Physical Name</th>
<th>Datatype</th>
<th>Length</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALES</td>
<td>StoreID</td>
<td>SQL_NUMERIC</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OldTransactionNumber</td>
<td>SQL_NUMERIC</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TransactionDate</td>
<td>SQL_DATE</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TransactionAmount</td>
<td>SQL_NUMERIC</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>REVENUE</td>
<td>StoreID</td>
<td>SQL_NUMERIC</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MonthlyRecognizedRevenue</td>
<td>SQL_NUMERIC</td>
<td>20</td>
<td>2</td>
</tr>
</tbody>
</table>

Table and column names are case sensitive.

2. To configure a filter that removes negative transactions, such as returns, enter the following expression in the Filters column for the SALES.TransactionAmount row:

   TransactionAmount > 0

3. To configure a lookup condition, enter the following expression in the Lookups column for the REVENUE.StoreID row:

   REVENUE.StoreID=SALES.StoreID

4. Save the changes.

The following figure shows the completed Source worksheet:

---

Configure the Target Worksheet

Use the Target worksheet to define the targets. You can also define aggregate and non-aggregate expressions for the mapping. In this example, you configure the following metadata:

- **Target table named T_SALES.** A target defined in the mapping specification becomes a target definition in a PowerCenter mapping. To configure a target port in the mapping specification, enter an expression in the Formula/Expression column for the target port. The expression can be the name of a source port or an aggregate or non-aggregate expression. Unconfigured target ports appear in a PowerCenter target definition but are not connected to the PowerCenter mapping pipeline. When defining target ports, use the target table name and column name as follows:

  `<TableName>.<ColumnName>`

- **Aggregate expression.** Use aggregate expressions to perform calculations on multiple values in a port. An aggregate expression defined in a mapping specification becomes an Aggregator transformation in a PowerCenter mapping. Define an aggregate expression in the row where you want return values to be written. This example uses the following aggregate expression to find the total sales from all transactions grouped by store ID:

  ```
  SUM( SALES.TransactionAmount )
  ```

  When you import a mapping specification, any port in the Aggregator transformation without an aggregate expression becomes a group by port. Use the Transformations Description column in the Target worksheet to indicate the group by ports you want to use. After importing the mapping specification, the PowerCenter
developer can view that information in the Aggregator transformation and configure the group by ports appropriately.

To configure the Target worksheet:

1. On the Target worksheet, enter the following information:

<table>
<thead>
<tr>
<th>Classes/Entities/Tables/Records Physical Name</th>
<th>Attributes/Columns/Fields Physical Name</th>
<th>Datatype</th>
<th>Length</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>T_SALES</td>
<td>StoreID</td>
<td>SQL_NUMERIC</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TransactionDate</td>
<td>SQL_DATE</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>StoreSales</td>
<td>SQL_NUMERIC</td>
<td>20</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>MonthlyRecognizedRevenue</td>
<td>SQL_NUMERIC</td>
<td>20</td>
<td>10</td>
</tr>
</tbody>
</table>

Table and column names are case sensitive.

2. To configure an aggregate expression that calculates total sales by store, enter the following expression in the Transformation Formula/Expression column for the T_SALES.StoreSales row:
   
   \[ \text{SUM(SALES.TransactionAmount)} \]

3. In the Description column, enter the following information for the PowerCenter developer:
   
   group by StoreID

4. To define the return value of the lookup condition of the REVENUE table, enter the following expression in the Transformation Formula/Expression column for the T_SALES.MonthlyRecognizedRevenue row:
   
   REVENUE.MonthlyRecognizedRevenue

5. To configure the remaining target ports, enter the following information:

<table>
<thead>
<tr>
<th>Attributes/Columns/Fields Physical Name</th>
<th>Transformations Formula/Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>StoreID</td>
<td>SALES.StoreID</td>
</tr>
<tr>
<td>TransactionDate</td>
<td>SALES.TransactionDate</td>
</tr>
</tbody>
</table>

6. Save the changes.

The following figure shows the completed Target worksheet:

---

**Step 3. Validate the Mapping Specification**

The Mapping template with validation includes macros to perform validation of the Source and Target worksheets.

To perform validation:

1. On the Source worksheet, click Validate.

   The validation notifies you of any error.
2. Correct errors and click Validate again.

3. On the Target worksheet, click Validate.
   The validation fills in the transformation types, “expr” for expression and “aggr” for aggregation and notifies you of any error.

4. Correct errors and click Validate again.

5. Save the changes.

Step 4. Save the Mapping Specification to a Shared Location

Save the mapping specification named AggregateMapping.xls to a shared location so that the PowerCenter developer can access the file. The PowerCenter developer imports the mapping specification into PowerCenter.

Step 5. Import the Mapping Specification

The PowerCenter developer uses the Repository Manager to import a Mapping Analyst for Excel mapping specification. The Repository Service exports mapping metadata from the mapping specification and then imports it to the PowerCenter repository as PowerCenter objects.

To import a mapping specification:

1. Copy the mapping specification named AggregateMapping.xls from the shared location to the following directory on your local machine:
   `<PowerCenter Client installation directory>\bin\Sample Specifications`

2. In the Repository Manager, open the folder that you selected to store the imported PowerCenter objects.

3. Click Repository > Import Metadata.
   The Tool Selection page appears.

4. For Source Tool, select Microsoft Office Excel and click Next.
   The Microsoft Office Excel Options page appears.

5. Enter the following information:

<table>
<thead>
<tr>
<th>Microsoft Excel Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>File name for the mapping specification. Click the Value field to locate and select the mapping specification that you created. For example, select AggregateMapping.xls.</td>
</tr>
<tr>
<td>With metadata mapping spreadsheet</td>
<td>Metamap associated with the template used to create the file. Click the Value field to locate and select the XML version of the associated metamap, MappingTemplate-MetaMap.xml.</td>
</tr>
</tbody>
</table>

Use the default values of the remaining options.

6. Click Next.
   The PowerCenter Options page appears.

7. For export objects, select Sources, Targets, and Mappings to import the entire mapping.
8. For Code Page, enter the name of the PowerCenter Repository Service code page.
For example, if the PowerCenter Repository Service uses the “MS Windows Latin 1 (ANSI), superset of Latin1” code page, enter the corresponding code page name MS1252. For a list of supported code page names, see the PowerCenter Administrator Guide.

9. Use the default values of the remaining options.

10. Click Next.

   In the Import Results dialog box, a message appears if the export from the mapping specification is successful. Error messages display when the export is not successful.

11. Click Show Details to view log events. You can also click Save Log to save the log events to a file.

12. Click Next.

   The Selection dialog box displays the sources or targets in the mapping specification with all objects selected by default.

13. Select all objects for import and click Finish.

   The Repository Service imports the mapping from the mapping specification. The results of the import appear in the Output window. By default, the mapping is named Mapping. You can change the mapping name when you edit the mapping.

You can view and edit the mapping in the Designer. The following figure shows the imported mapping:

![Imported Mapping Diagram]

---

**Step 6. Edit the PowerCenter Mapping and Generate a Workflow**

The PowerCenter developer completes the following tasks in the PowerCenter Client:

♦ Edit the PowerCenter mapping.
♦ Generate and run a workflow from the mapping.

**Edit the PowerCenter Mapping**

The PowerCenter developer uses the Designer to view and edit the PowerCenter mapping. The developer can implement additional functionality that might be required. For this example, edit the mapping to configure the Aggregator transformation group by ports.

**To edit the mapping to configure the Aggregator transformation group by ports:**

1. In the Mapping Designer, open the mapping for editing.

2. To rename the mapping, click Mapping > Edit, enter a name, and click OK.
   For example, enter AggregateMappingExample.

3. Double-click the Aggregator transformation.

4. On the Ports tab, click the StoreSales port.
   The Description field displays the note to configure StoreID as the group by port.

5. Select the GroupBy option for the StoreID port, and clear the option for all other ports.
6. Click OK.
7. Save the mapping.

Generate and Run a Workflow from the Mapping

The PowerCenter developer uses the Workflow Generation Wizard in the Designer to generate a workflow from the mapping. The developer then runs the workflow to populate the target table. For more information about using the Workflow Generation Wizard, see the *PowerCenter Designer Guide*.

After the workflow runs, use Oracle SQL Developer to review the contents of the target table. For example, run the following SQL statement to view the contents of the table:

```sql
SELECT * FROM dm.T_Sales ORDER BY storeID;
```
Overview

A business analyst uses the Source-Target-Matrix template to create a mapping specification for multiple mappings. The Source-Target-Matrix template contains a separate worksheet for each mapping in the mapping specification. Mapping specifications based on the Source-Target-Matrix template can contain sources, targets, and Expression transformations. A PowerCenter developer imports the mapping specification using the Repository Manager to create the corresponding PowerCenter objects.

To create a mapping specification based on the Source-Target-Matrix template, a business analyst completes the following steps:

1. Copy and rename the template.
2. Configure the name for each mapping.
3. Configure sources.
4. Configure Expression transformations.
5. Configure targets.

Note: You can use the Source-Target-Matrix template to create a mapping specification with a single mapping.

Copying and Renaming the Source-Target-Matrix Template

To create a mapping specification based on the Informatica Source-Target-Matrix template, copy and rename the template. You can find the Source-Target-Matrix template, source-target-matrix.xls, in the following directory:

\<PowerCenter Client installation directory>\bin\Sample Specifications
Mapping Names

The Source-Target-Matrix template contains three worksheets named mapping1, mapping2, and mapping3. When a PowerCenter developer imports a mapping specification created from the Source-Target-Matrix template, the Repository Service performs the following actions:

- Names each mapping and target in the mapping specification with the worksheet name. Rename each worksheet with an appropriate mapping and target name.
- Creates a new mapping for every worksheet in the file. To avoid importing unused worksheets, delete all unused worksheets from the file.

Sources

Use the Source section on each worksheet to enter source information for the mapping.

Enter the following properties in the Source section:

<table>
<thead>
<tr>
<th>Property</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File/table</td>
<td>Required</td>
<td>File or table name. The Repository Service creates a PowerCenter source definition with this name.</td>
</tr>
<tr>
<td>Field/column</td>
<td>Required</td>
<td>Field or column name. The Repository Service creates a port in the source definition with this name. Add related fields and columns in the same order as in the Target section on the worksheet. When you import the mapping specification, the Reporting Service links transformation ports by position. For example, the Reporting Service links the first output port to the first input port, and the second output port to the second input port.</td>
</tr>
<tr>
<td>Field/column Datatype</td>
<td>Required</td>
<td>Datatype of the field or column. For sources in relational databases, enter the Mapping Analyst for Excel datatype. The Repository Service assigns the corresponding native datatype to the port in the source definition. For more information, see “Mapping Analyst for Excel Datatypes” on page 8.</td>
</tr>
<tr>
<td>Field/column Description</td>
<td>Optional</td>
<td>Description for the field or column. The Repository Service uses this description for the port in the source definition.</td>
</tr>
</tbody>
</table>

To specify a database type for the source, the PowerCenter developer can complete one of the following tasks:

- **Select a database type when importing the mapping specification.** On the PowerCenter Options page of the Import Wizard, select a database type for the Database type option. During import, the Repository Service assigns this database type to all sources and targets for each mapping configured in the mapping specification. For more information, see “Importing Mapping Specifications” on page 34.

- **Edit the database type for the source definition in the Designer.** After importing the mapping specification, open the source definition in the Source Analyzer. In the Table tab, select the database type.

To specify a database name for the source, the PowerCenter developer can enter the database name when importing the mapping specification. On the PowerCenter Options page of the Import Wizard, enter a name for the Database name option. All sources for each mapping configured in the mapping specification will have the same database name. For more information, see “Importing Mapping Specifications” on page 34.
Expression Transformations

Mapping specifications based on the Source-Target-Matrix template can contain non-aggregate expressions. Use an expression to calculate values in a single row for each row in a port. For example, the following expression increases the salary of each employee by seven percent:

\[ \text{Salary} + \left( \text{Salary} \times 0.07 \right) \]

You can use conditional clauses and non-aggregate functions in non-aggregate expressions. You can also nest non-aggregate functions.

You configure an expression in the Transformation column for the source row on each worksheet of the mapping specification. An expression defined in the mapping specification becomes an Expression transformation in a PowerCenter mapping.

The following figure shows an expression defined for a source row named Salary:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data type</th>
<th>Length</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>EmpID</td>
<td>EmpID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>First Name</td>
<td>First Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Last Name</td>
<td>Last Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Salary</td>
<td>Salary</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Start Date</td>
<td>Start Date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>End Date</td>
<td>End Date</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Targets

Use the Target section on each worksheet to enter target information for the mapping.

When a PowerCenter developer imports the mapping specification, the Repository Service uses the worksheet name to define the target definition name.

Enter the following properties in the Target section:

<table>
<thead>
<tr>
<th>Property</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field/column</td>
<td>Required</td>
<td>Field or column name. The Repository Service creates a port in the target definition with this name. Add related fields and columns in the same order as in the Source section on the worksheet. When you import the mapping specification, the Reporting Service links transformation ports by position. For example, the Reporting Service links the first output port to the first input port, the second output port to the second input port, and so forth.</td>
</tr>
<tr>
<td>Field/column Datatype</td>
<td>Required</td>
<td>Datatype of the field or column. Enter the datatype in either column. For targets in relational databases, enter the Mapping Analyst for Excel datatype. The Repository Service assigns the corresponding native datatype to the port in the target definition. For more information, see “Mapping Analyst for Excel Datatypes” on page 8.</td>
</tr>
<tr>
<td>Length</td>
<td>Required</td>
<td>Length of the column. The Repository Service assigns this value to the precision of the port in the target definition.</td>
</tr>
<tr>
<td>Scale</td>
<td>Required</td>
<td>Scale of the column. The Repository Service assigns this value to the scale of the port in the target definition.</td>
</tr>
</tbody>
</table>
To specify a database type for the target, the PowerCenter developer can complete one of the following tasks:

- **Select a database type when importing the mapping specification.** On the PowerCenter Options page of the Import Wizard, select a database type for the Database type option. During import, the Repository Service assigns this database type to all sources and targets for each mapping configured in the mapping specification. For more information, see “Importing Mapping Specifications” on page 34.

- **Edit the database type for the target definition in the Designer.** After importing the mapping specification, open the target definition in the Target Designer. In the Table tab, select the database type.

<table>
<thead>
<tr>
<th>Property</th>
<th>Required/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Optional</td>
<td>Description for the field or column. The Repository Service uses this description for the port in the target definition.</td>
</tr>
<tr>
<td>Notes</td>
<td>Optional</td>
<td>Description for the field or column. The Repository Service uses this description for the port in the target definition. When you enter values for both the Description and Notes properties, the Repository Service uses the Notes property.</td>
</tr>
</tbody>
</table>
Configuring Multiple Mappings - Example

This chapter includes the following topics:
♦ Overview, 27
♦ Step 1. Establish the PowerCenter Environment, 28
♦ Step 2. Configure the Mapping Specification in Microsoft Excel, 28
♦ Step 3. Save the Mapping Specification to a Shared Location, 30
♦ Step 4. Import the Mapping Specification, 30
♦ Step 5. Edit the PowerCenter Mappings and Generate Workflows, 31

Overview

This example uses the Source-Target-Matrix mapping specification template to configure mappings that read human resources data from a flat file and write the data to an Oracle database. The mapping specification in this example configures the following mappings:
♦ **Employee_Info.** Reads a delimited flat file containing employee information and writes the data to a target table named Employee_Info located in an Oracle schema named DM.
♦ **Emp_Raises.** Reads the same delimited flat file, includes an Expression transformation that calculates a salary increase for each employee, and writes the data to a target table named Emp_Raises located in an Oracle schema named DM.

To create these example mappings, the business analyst and PowerCenter developer complete the following steps:
1. Establish the PowerCenter environment (Developer). The PowerCenter developer configures the database connectivity, designates a repository folder to store the imported objects, and creates the sample flat file source and the target tables.
2. Configure the mapping specification in Microsoft Excel (Business Analyst). The business analyst creates a mapping specification from the Source-Target-Matrix template and configures the Employee_Info and Emp_Raises mappings on separate worksheets.
3. Save the mapping specification to a shared location (Business Analyst).
4. Import the mapping specification (Developer).
5. Edit the PowerCenter mappings and generate workflows (Developer).
Step 1. Establish the PowerCenter Environment

Before the business analyst creates a mapping specification, the PowerCenter developer completes the following tasks to establish the PowerCenter environment:

- **Configure database connections.** Use the Workflow Manager to configure connection objects so that the Integration Service can connect to the source and target databases.
- **Designate a repository folder to store the imported objects.** Use the Repository Manager to select an existing folder or to create a folder for the PowerCenter objects that are imported from the mapping specification.
- **Create the source flat file.** Create a flat file that contains the sample data used in this example.
- **Create target tables.** Use Oracle SQL Plus to run SQL statements to use the sample data in this example.

Create the Source Flat File

To use the source flat file included in this example, create a sample flat file that contains the following data:

- EmployeeID
- FirstName
- LastName
- Salary
- StartDate
- EndDate

For example, create a file named HR.txt that contains the following sample data:

```
1,John,Baer,100000,1/1/2005 00:00:00,1/1/2006 00:00:00
2,Bobbi,Apperley,120000,1/1/2005 00:00:00,1/1/2007 00:00:00
```

Save the file to the source file directory for the Integration Service process. The directory is defined in the $PMSourceFileDir service process variable. By default, the variable is set to the following directory:

<PowerCenter installation directory>\server\infa_shared\SrcFiles

Create Target Tables

To use the target tables included in this example, use Oracle SQL Plus to run the following SQL statements to create the Employee_Info and Emp_Raises target tables in an Oracle schema named DM:

```
Connect DM/DM
CREATE TABLE DM.Employee_Info (Emp_ID number, First_Name varchar2(40), Last_Name varchar2(40), Salary number, ST_Date date, End_Date date);
CREATE TABLE DM.Emp_Raises (Emp_ID number, First_Name varchar2(40), Last_Name varchar2(40), Salary number, ST_Date date, End_Date date);
```

Step 2. Configure the Mapping Specification in Microsoft Excel

The business analyst completes the following tasks to configure the mapping specification in Microsoft Excel:

- Create a mapping specification based on the Source-Target-Matrix template.
- Configure the mappings.
Create a Mapping Specification

To create a mapping specification based on the Informatica Source-Target-Matrix template, copy and rename the template. You can find the Source-Target-Matrix template, source-target-matrix.xls, in the following directory:

<PowerCenter Client installation directory>\bin\Sample Specifications

For example, copy and rename the template HR_Data_Load.xls.

Configure the Mappings

Configure each mapping on a different worksheet in the mapping specification and name each worksheet. The Source-Target-Matrix template uses the worksheet name for both mapping names and target names in the mapping.

To configure multiple mappings:

1. Open HR_Data_Load.xls.
2. Rename the first worksheet Employee_Info and press Enter.
3. In the Source section of the Employee_Info worksheet, enter the following information:

<table>
<thead>
<tr>
<th>File/Table</th>
<th>Field/Column</th>
<th>Field/Column Datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>HR</td>
<td>EmployeeID</td>
<td>Number</td>
</tr>
<tr>
<td></td>
<td>FirstName</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td>LastName</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td>Salary</td>
<td>Decimal</td>
</tr>
<tr>
<td></td>
<td>StartDate</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td>EndDate</td>
<td>Date</td>
</tr>
</tbody>
</table>

4. In the Target section of the Employee_Info worksheet, enter the following information:

<table>
<thead>
<tr>
<th>Field/Column</th>
<th>Field/Column Datatype Native Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp_ID</td>
<td>SQL_INTEGER</td>
</tr>
<tr>
<td>First_Name</td>
<td>SQL_CHAR</td>
</tr>
<tr>
<td>Last_Name</td>
<td>SQL_CHAR</td>
</tr>
<tr>
<td>Salary</td>
<td>SQL_DECIMAL</td>
</tr>
<tr>
<td>ST_Date</td>
<td>SQL_DATE</td>
</tr>
<tr>
<td>End_Date</td>
<td>SQL_DATE</td>
</tr>
</tbody>
</table>

5. Save the changes.
6. To create the second mapping, copy rows 3 through 8 from the Employee_Info worksheet to the second worksheet.
7. Rename the second worksheet Emp_Raises and press Enter.
8. In the Salary row, enter the following expression in the Transformation column to define a seven percent increase in salary for all employees:

   Salary + (Salary * .07)

9. Save the changes.
10. Delete the unused worksheet named mapping3.
Step 3. Save the Mapping Specification to a Shared Location

Save the mapping specification named HR_Data_Load.xls to a shared location so that the PowerCenter developer can access the file. The PowerCenter developer imports the mapping specification into PowerCenter.

Step 4. Import the Mapping Specification

The PowerCenter developer uses the Repository Manager to import a Mapping Analyst for Excel mapping specification. The Repository Service exports mapping metadata from the mapping specification and then imports it to the PowerCenter repository as PowerCenter objects.

To import a mapping specification:

1. Copy the mapping specification named HR_Data_Load.xls from the shared location to the following directory on your local machine:
   \<PowerCenter Client installation directory>\bin\Sample Specifications

2. In the Repository Manager, open the folder that you selected to store the imported PowerCenter objects.

3. Click Repository > Import Metadata.
   The Tool Selection page appears.

4. For Source Tool, select Microsoft Office Excel and click Next.
   The Microsoft Office Excel Options page appears.

5. Enter the following information:

<table>
<thead>
<tr>
<th>Microsoft Excel Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>File name for the mapping specification. Click the Value field to locate and select the mapping specification that you created. For example, select HR_Data_Load.xls.</td>
</tr>
<tr>
<td>With metadata mapping spreadsheet</td>
<td>Metamap associated with the template used to create the file. Click the Value field to locate and select the XML version of the associated metamap, source-target-matrix-metamap.xml.</td>
</tr>
</tbody>
</table>

   Use the default values of the remaining options.

6. Click Next.
   The PowerCenter Options page appears.

7. For export objects, select Sources, Targets, and Mappings to import the entire mapping.

8. For Code Page, enter the name of the PowerCenter Repository Service code page.
   For example, if the PowerCenter Repository Service uses the “MS Windows Latin 1 (ANSI), superset of Latin1” code page, enter the corresponding code page name MS1252. For a list of supported code page names, see the PowerCenter Administrator Guide.

   Use the default values of the remaining options.
9. Click Next.
   In the Import Results dialog box, a message appears if the export from the mapping specification is successful. Error messages display when the export is not successful.

10. Click Show Details to view log events. You can also click Save Log to save the log events to a file.

11. Click Next.
   The Selection dialog box displays the sources or targets in the mapping specification with all objects selected by default.

12. Select all objects for import and click Finish.
   The Repository Service imports the mapping from the mapping specification. The results of the import appear in the Output window. By default, the Repository Service uses the worksheet names for the mapping names. You can change the mapping names when you edit the mappings.

You can view and edit the mappings in the Designer. The following figure shows the imported Employee_Info mapping:

![Employee_Info mapping](image)

The following figure shows the imported Emp_Raises mapping:

![Emp_Raises mapping](image)

Step 5. Edit the PowerCenter Mappings and Generate Workflows

The PowerCenter developer completes the following tasks in the PowerCenter Client:

- Edit the PowerCenter mappings.
- Generate and run workflows from the mappings.

Edit the PowerCenter Mappings

The PowerCenter developer uses the Designer to view and edit the PowerCenter mapping. The developer can implement additional functionality that might be required. For this example, edit the source definition to change the flat file format to delimited.

**To edit the mappings to specify a delimited flat file format:**

1. In the Source Analyzer, open the HR source definition for editing.
2. On the Table tab, select Delimited for Flat File Information.
3. Click OK.
4. Save the source definition.

Generate and Run Workflows from the Mappings

The PowerCenter developer uses the Workflow Generation Wizard in the Designer to generate workflows from the mappings. The developer then runs the workflows to populate the target tables. For more information about using the Workflow Generation Wizard, see the PowerCenter Designer Guide.

After the workflows run, use Oracle SQL Developer to review the contents of the target tables. For example, run the following SQL statements to view the contents of the tables:

```sql
Select * from dm.Employee_Info;
Select * from dm.Emp_Raises;
```

Compare the Salary column in both tables to review the salary increases.
Overview

You can use the PowerCenter Repository Manager to import the following information from a mapping specification:

- Source definitions
- Target definitions
- A single mapping
- Multiple mappings that use the same mapping specification template

For example, a business analyst creates a mapping specification in Microsoft Excel to define a mapping that includes sources, targets, and transformations. A PowerCenter developer imports the mapping specification to create the PowerCenter objects.

You can use the PowerCenter Repository Manager to export the following information from a PowerCenter repository to a mapping specification:

- Source definitions
- Target definitions
- A single, valid mapping

You might want to export PowerCenter objects to Microsoft Excel for the following reasons:

- Several workflows are running in production. However, no documentation about these workflows or mappings exists. A PowerCenter developer can export the mappings to mapping specifications for documentation and business analyst review.
- You are using Mapping Analyst for Excel to configure PowerCenter mappings, and the business analyst needs to review the implementation of their mapping specifications. A PowerCenter developer can export the mapping for a business analyst to review at any point in the development process.
**Importing Mapping Specifications**

Use the Repository Manager to import a mapping specification. You can import sources and targets, or you can import an entire mapping. You can also import more than one mapping at a time if the mapping specification contains multiple mappings. Mappings do not have to be complete or valid.

When you import a mapping specification, the Repository Service exports metadata from the mapping specification and imports it into the PowerCenter repository.

You can use the Mapping template, the Source-Target-Matrix template, or a custom template to create a mapping specification. The following table describes the PowerCenter objects that the Repository Service creates when you import a mapping specification based on the Mapping or Source-Target-Matrix template:

<table>
<thead>
<tr>
<th>Mapping Analyst for Excel Component</th>
<th>Template</th>
<th>PowerCenter Repository Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source information.</td>
<td>Mapping</td>
<td>Source definitions.</td>
</tr>
<tr>
<td>Filter expression to filter data from the pipeline.</td>
<td>Mapping</td>
<td>Filter transformation.</td>
</tr>
<tr>
<td>Join expression to join two sources. To join more than two sources, you can enter multiple join expressions.</td>
<td>Mapping</td>
<td>Joiner transformation.</td>
</tr>
<tr>
<td>Lookup expression to perform a lookup on source data.</td>
<td>Mapping</td>
<td>Connected Lookup transformation. Includes ports that are used in the mapping.</td>
</tr>
<tr>
<td>Aggregate data transformation expression.</td>
<td>Mapping</td>
<td>Aggregator transformation.</td>
</tr>
<tr>
<td>Single row data transformation expression.</td>
<td>Mapping</td>
<td>Expression transformation.</td>
</tr>
<tr>
<td>Target information.</td>
<td>Mapping</td>
<td>Target definitions.</td>
</tr>
<tr>
<td>Additional comments or descriptions</td>
<td>Mapping</td>
<td>Descriptions associated with transformations and source and target definitions. Provides additional notes to the PowerCenter developer.</td>
</tr>
</tbody>
</table>

After you import a mapping specification, review the imported objects. Most mappings imported from a mapping specification require editing before validation.

**To import a mapping specification:**

1. In the Repository Manager, open a folder and click Repository > Import Metadata.
   The Tool Selection page appears.

2. In the Source Tool field, select Microsoft Office Excel and click Next.
   The Microsoft Office Excel Options page appears.

3. Enter the following information:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>File name for the mapping specification. Click the Value field to locate and select the mapping specification to import.</td>
</tr>
</tbody>
</table>
4. Use the default values of the remaining options.
5. Click Next.
   The PowerCenter Options page appears.
6. Enter the following information:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>With metadata mapping spreadsheet</td>
<td>Metamap associated with the template used to create the mapping specification. Click the Value field to locate and select the XML version of the associated metamap.</td>
</tr>
<tr>
<td>Environment variable file</td>
<td>Environment variable file that contains properties for the metamap. Click the Value field to locate and select an environment variable file. Optional.</td>
</tr>
</tbody>
</table>

7. Click Use Defaults to revert to default options.
8. Click Next.
   In the Import Results dialog box, a message appears if the export from the mapping specification is successful. Error messages appear when the export is not successful.
   Click Show Details to view log events. You can also click Save Log to save the log events to a file.
9. Click Next.
   The Source Selection dialog box displays the sources or targets in the mapping specification with all objects selected by default.
10. Select the objects that you want to import and click Finish.
    The Repository Service imports the metadata from the mapping specification.
    If the import folder contains objects of the same name as those you are importing, the Conflict Resolution Wizard displays. For more information about the conflict, click Compare Conflict.
    Otherwise for each conflicting object, select the object and choose one of the following resolutions:
     ♦ Rename. Change the object name when importing it to the target folder.
You can also apply the same resolution to all sources or to all conflicts.

11. Click Next, and then click Close to close the Conflict Resolution Wizard.

### Exporting Mappings

Use the Repository Manager to export objects from the PowerCenter repository to a mapping specification. You can export sources and targets, or you can export a valid mapping.

Informatica provides mapping specification templates that organize the mapping metadata in different ways. When you export a mapping to Microsoft Excel, use one of the following mapping specification templates:

- **Mapping template (with validation).** Contains a single mapping configured on multiple worksheets for overview information, sources, targets, and functions. Mapping specifications based on this template can contain source and target data and Joiner, Lookup, Filter, Expression, and Aggregator transformations. This template performs some validation of the metadata.

- **Mapping template (without validation).** Is the same as the Mapping template (with validation). However, this template does not perform validation of the metadata. Use this template if you want to avoid enabling macros on your machine.

- **Source-Target-Matrix template.** Can contain multiple mappings, each configured on a separate worksheet. Mapping specifications based on this template can contain source and target data and Expression transformations. If the mapping being exported contains additional transformations, use the Mapping template.

- **Custom template.** Contains a custom format. You can create a custom mapping specification template if you want to use a format different from the Mapping Analyst for Excel templates.

The Repository Service exports only supported transformations. When you export a mapping that includes other transformations, target definitions in the same pipeline might not contain all ports in the mapping specification.

You can export a Lookup transformation that performs a lookup on a source definition. The Repository Service does not export a Lookup transformation based on a lookup source that is not defined in the mapping.

**To export PowerCenter repository objects to a mapping specification:**

1. In the Repository Manager, open a folder that contains the repository objects that you want to export.
2. In the Navigator, select the mapping or set of sources or targets that you want to export.
3. Click Repository > Export Metadata.
   The Tool Selection page appears.
4. For Target Tool, select Microsoft Office Excel and click Next.
   The Microsoft Office Excel Options page appears.
5. Enter the following information:

<table>
<thead>
<tr>
<th>Microsoft Excel Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td>Name of the mapping specification for exported metadata. Click the Value field to locate and select the mapping specification you want to use. Or, enter a file name for the wizard to create. You can export metadata to the same file more than once. When you export to a mapping specification that contains existing data, the Repository Service writes information to empty cells or overwrites existing data.</td>
</tr>
<tr>
<td><strong>With metadata mapping spreadsheet</strong></td>
<td>Metamap associated with the mapping specification template you want to use. Click the Value field to locate and select one of the following metamsaps: - MappingTemplate-MetaMap.xml. Select if using the Mapping template with or without validation. Exports the mapping to a Microsoft Excel file with multiple worksheets for overview information, sources, targets, and functions. - source-target-matrix-metamap.xml. Select if using the Source-Target-Matrix template. Exports the mapping to a Microsoft Excel file on a single worksheet. - &lt;custom-metamap&gt;.xml. If using a custom mapping specification template, select the name of the associated custom metamap.</td>
</tr>
<tr>
<td><strong>From base spreadsheet</strong></td>
<td>Mapping specification template. Click the Value field to locate and select one of the following mapping specification template files: - MappingTemplate.xls. Select if using the Mapping template and you want to validate the metadata. - MappingTemplate_NoValidation.xls. Select if using the Mapping template and you do not want to enable macros to validate the metadata. - source-target-matrix-mirrepeatingsheet.xls. Select if using the Source-Target-Matrix template. - &lt;custom-template&gt;.xls. If using a custom mapping specification template, select the name of the custom template.</td>
</tr>
</tbody>
</table>

*Note: Use the default values of the remaining options.*

6. Click Use Defaults to reset to default values.

7. Click Export.

In the Export Results dialog box, a message appears if the export is successful. Error messages display when the export is not successful.

Click Show Details to view log events. You can also click Save Log to save the log events to a log file.

8. Click Finish.

Troubleshooting

I have several mappings in a single file. Each mapping uses the same template and is on a single worksheet. However, I cannot import the mapping specification.

Verify that the Source-Target-Matrix repeating sheet file and metamap are configured correctly. The metamap must reference fields in the Source-Target-Matrix repeating sheet file.

I receive the following error when I import a mapping specification:

This workbook contains links to other data sources....

The metamap references an unrelated mapping specification template. Update the metamap and import the mapping specification again.

The following error message appears in the log when I import a mapping specification:

To avoid potential complications with metadata mapping, it is strongly recommended to use metadata mapping file in XML format.
When you imported the mapping specification, you selected a metamap in Microsoft Excel format. Save the metamap as an XML file and import the mapping specification again.

I used the Mapping template to export a mapping that contains an Aggregator transformation. However, the exported mapping specification does not include the aggregate expression on the Target worksheet.

In the Designer, verify that the expression port in the Aggregator transformation is connected to a mapping target. To export an aggregate expression to a mapping specification, the expression port in the Aggregator transformation must be connected to a mapping target.
Overview

Create a custom mapping specification template and associated metamap if you want to use a format different from the Mapping Analyst for Excel templates. After you configure a custom mapping specification template and the associated metamap, you can use the custom mapping specification template to configure a mapping specification.

To create a custom mapping specification template and associated metamap, complete the following steps:

1. Configure a custom mapping specification template.
2. Copy and rename the metamap.
3. Associate the metamap with the custom mapping specification template or repeating sheet file.
4. Configure the Model, Source, Target, and Mapping worksheets in the metamap.
5. Optionally configure import and export parameters in the metamap.
6. Save the metamap as an XML spreadsheet.

Metamap Hierarchy

The metamap divides the mapping specification metadata into a hierarchy for sources, targets, transformations, and mappings. The metamap contains a worksheet for each object in the hierarchy. When you import the mapping specification, the Repository Service imports the metadata in the file starting at the top of the hierarchy.
The following figure shows the Mapping Analyst for Excel metadata hierarchy:

The metamap hierarchy contains parent and child objects. If you configure a child object, you must also configure the parent object on its corresponding worksheet. For example, a source schema is a child object of a source data package and a model. If you configure a source schema, you also must configure a data package and a model.

Each metamap worksheet describes metadata for objects at that level and identifies all parent objects. For example, the SourceSchemas worksheet describes the source schema and identifies its parent source data package and parent model.

When you modify the associated metamap to reflect the changes made to a custom mapping specification template, you edit the metamap worksheets for the metadata type that you modified. For example, if you modify the source metadata in a custom version of the Mapping template, you modify the source and mapping worksheets in the metamap. You do not modify the target worksheets.

Mapping Analyst for Excel requires you to create some objects that are part of the metadata hierarchy that PowerCenter does not import. For example, the Repository Service does not import the Model and Data Package name. You can enter constant values for these objects in the metamap if the mapping specification does not contain values.

The following table describes the metamap worksheet types and how the Repository Service reads each worksheet type:

<table>
<thead>
<tr>
<th>Worksheet Type</th>
<th>Worksheet Names</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>Overview</td>
<td>Determines how the Repository Service imports and exports metadata and the template type you want to use.</td>
</tr>
<tr>
<td>Model</td>
<td>Models</td>
<td>Defines the location of the entity that contains all mapping metadata. The Repository Service does not import this information.</td>
</tr>
<tr>
<td>Source</td>
<td>SourceDataPackages</td>
<td>Defines the location of source metadata. The Repository Service reads each worksheet in the hierarchy to create a source definition.</td>
</tr>
</tbody>
</table>
Rules and Guidelines

Use the following rules and guidelines when you create a custom mapping specification template:

♦ Decide on a mapping structure before you design the template. One or two templates and associated metamaps are more efficient than multiple templates and associated metamaps for each mapping.

♦ Create separate fields for different types of metadata. This allows you to match each field to different parts of the metamap.

♦ Configure names for the mapping, sources, targets, or transformations in the mapping specification or as constants in the metamap.

♦ Use a single worksheet to import more than one mapping at a time.

♦ Edit the metamap to reflect the custom mapping specification template structure. To avoid confusion, complete the template design before you configure the metamap.

Configuring a Custom Mapping Specification Template

Use one of the following methods to configure a custom mapping specification template:

♦ Copy, rename, and modify the Mapping template. Copy and rename MappingTemplate.xls or MappingTemplate_NoValidation.xls, depending on whether you want to use a template with or without validation. Use Microsoft Excel to modify the renamed mapping specification template by adding or deleting columns.

♦ Copy, rename, and modify the Source-Target-Matrix template and repeating sheet file. Copy and rename source-target-matrix.xls and source-target-matrix-mirrepeatingsheet.xls. Use Microsoft Excel to modify the renamed mapping specification template and repeating sheet file by adding or deleting columns.

The repeating sheet file defines the common structure for multiple mappings defined in a mapping specification that is based on the Source-Target-Matrix template. When you modify a custom version of the Source-Target-Matrix template, you must make the same changes to a custom version of the Source-Target-Matrix repeating sheet file. The Source-Target-Matrix metamap uses the Source-Target-Matrix repeating sheet file to define the location of metadata in the Source-Target-Matrix mapping specification template.

♦ Create a new template. Use Microsoft Excel to create a new spreadsheet if the metadata format that you want to use differs significantly from the Informatica mapping specification templates.
The original mapping specification templates and the Source-Target-Matrix repeating sheet file are located in the following directory:

\<PowerCenter Client installation directory>\bin\Sample Specifications

**Note:** A metamap uses relative links to find the location of metadata in a mapping specification. To avoid broken links, use Windows Explorer to copy the templates. Do not use the Save As menu option in Microsoft Excel.

### Copying and Renaming the Metamap

Copy and rename the .xls version of an existing metamap. Mapping Analyst for Excel includes metamaps in Excel and XML formats. Use the Excel version of the metamap to modify metamap properties when you create a custom mapping specification template.

If your custom mapping specification template is based on an existing template, copy and rename the associated metamap for the existing template: MappingTemplate-MetaMap.xls or source-target-matrix-metamap.xls.

If you created a new custom mapping specification template, copy and rename either of the existing metamaps. When you create a new mapping specification template, you must modify each worksheet in a metamap. As a result, you can use any existing metamap to create the custom version.

The original files are located in the following directory:

\<PowerCenter Client installation directory>\bin\Sample Specifications

**Note:** A metamap uses relative links to find the location of metadata in a mapping specification. To avoid broken links, use Windows Explorer to copy the templates. Do not use the Save As menu option in Microsoft Excel.

### Associating the Metamap with the Custom Template or Repeating Sheet File

Use Microsoft Excel to associate a custom metamap with the file that the metamap needs to read. The file that you associate depends on the following existing metamaps that you used to create the custom metamap:

- **Mapping metamap.** Reads from the Mapping template to determine the structure of the mapping. When you rename the Mapping template, you must associate the metamap with the new name to maintain the metamap links to the mapping specification template.
- **Source-Target-Matrix metamap.** Reads from the Source-Target-Matrix repeating sheet file to determine the structure of the mapping on each worksheet of a mapping specification that is based on the Source-Target-Matrix template. When you rename the repeating sheet file, you must associate the metamap with the new repeating sheet file name to maintain the metamap links to the repeating sheet file.

**To associate the metamap with the mapping specification template or repeating sheet file:**

1. Open the custom mapping specification template or repeating sheet file.
2. Open the custom metamap.
   
   If a security warning appears to warn against using macros, click Enable Macros to enable the validation macros.
3. When a message box states that the metamap contains links to other data sources, click Update.
   
   A message states that one or more links cannot be updated.
4. Click Edit Links.
The Edit Links dialog box appears.

5. Click Change Source.

6. Select the custom mapping specification template or repeating sheet file, and click OK.

7. Click Close to close the Edit Links dialog box.

8. Save the metamap.

Model, Source, Target, and Mapping Worksheet Types in the Metamap

Edit the metamap worksheets for the metadata type that you modified in the mapping specification template. For example, if you modify the source metadata in a custom version of the Mapping template, you modify the source and mapping worksheets in the metamap. You do not modify the target worksheets.

Each metamap worksheet for the model, sources, targets, and mappings contains the following sections that you configure:

♦ Excel Table Parameters. First and last lines to process in the mapping specification template.

♦ Required Properties. The location of metadata in the mapping specification template.

♦ User-Defined Properties. The location of user-defined properties in the mapping specification template.

When you configure each section, you modify the location of metadata in the mapping specification template.

Note: When Hide Special Fields is selected in a metamap worksheet, the worksheet hides internal cells needed for proper functioning of the metamap. If you clear this option, do not modify the internal cells.

Excel Table Parameters

When you import a mapping specification, the Repository Service processes lines on each worksheet from top to bottom until it reaches the last line. The Excel Table Parameters section on each model, sources, targets, and mappings worksheet in the metamap defines the first and last line to process in a mapping specification worksheet.

For example, a mapping specification has a column that contains source attribute names. The attribute names start in column B, row 4. On the SourceAttributes worksheet in the metamap, configure the first line to process as an Exact Line starting at row 4.

When you enter constants in a metamap worksheet, you can set the first and last lines to be column A, row 1. The Repository Service processes only one line when you import the mapping specification.
On each worksheet for the model, sources, targets, and mappings in the metamap, configure the following Excel Table Parameters for the first and last lines:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| How to Detect      | Determines when the Repository Service starts or stops processing mapping specification lines for an object. Select one of the following options: 
- Exact Line. The Repository Service starts or stops processing when it reads a specific line. Default is Exact Line for the first line.
- First Empty Line. The Repository Service stops processing when it reads the first empty line.
- Count of Empty Lines. The Repository Service stops processing when it reads a number of empty lines. |
| Detection Parameter| Indicates a cell or line where the Repository Service starts or stops processing the mapping specification. Enter one of the following options: 
- Cell Location. The location of a cell where the Repository Service starts or stops processing lines on the mapping specification worksheet. Required if the processing is configured for an exact line. To add the cell location, see “Modifying the Cell Location of Metadata” on page 46.
- Number of Lines. The number of empty lines to read before the Repository Service stops processing the mapping specification worksheet. Required if the processing is configured to stop when a number of empty lines are read. |

**Required Properties**

When you configure the required properties, you define the location of the metadata in the mapping specification template.

The Repository Service reads the metamap and imports the metadata from the mapping specification according to the metamap hierarchy. The metamap hierarchy contains parent and child objects. The Required Properties section on each metamap worksheet describes metadata for objects at that level and identifies all parent objects.

For example, the SourceSchemas worksheet contains a data package and model name for the schema. The data package and model are the parent objects in the metadata hierarchy. If the mapping specification contains more than one data package, the Repository Service chooses the first data package name from the mapping specification as the parent.

The following figure shows the parent object properties configured in the SourceSchemas worksheet:
The following table describes the parameters for the required properties:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Style</td>
<td>Determines whether metadata occurs in more than one cell in the mapping specification template. Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Column. Metadata occurs more than once in a column of the mapping specification template. When you configure a reference to a column, the Repository Service processes each row in the column. The column can have empty rows.</td>
</tr>
<tr>
<td></td>
<td>- Cell. Metadata is in one cell in the mapping specification template.</td>
</tr>
<tr>
<td>Metadata field</td>
<td>Name of the metadata property. For a list of the metadata fields on each worksheet, see “Metamap Worksheets” on page 59.</td>
</tr>
<tr>
<td>If Empty</td>
<td>Determines how the Repository Service processes a blank metadata property. Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Print a warning and then take a default action. The Repository Service uses the default value or the default parent.</td>
</tr>
<tr>
<td></td>
<td>- Skip the row. The Repository Service does not create metadata for the metadata field.</td>
</tr>
<tr>
<td></td>
<td>- Stop the import. The Repository Service does not import the mapping specification into the repository.</td>
</tr>
<tr>
<td>Reference to Data</td>
<td>Contains a constant value or the location of a cell in the mapping specification template. To add the cell location, see “Modifying the Cell Location of Metadata” on page 46.</td>
</tr>
<tr>
<td>Validation</td>
<td>Displays one of the following values based on the Reference to Data parameter:</td>
</tr>
<tr>
<td></td>
<td>- Constant. The Reference to Data column contains a constant value.</td>
</tr>
<tr>
<td></td>
<td>- Reference. The Reference to Data column contains a cell location.</td>
</tr>
<tr>
<td></td>
<td>- Invalid. The Reference to Data column contains invalid values.</td>
</tr>
<tr>
<td></td>
<td>This parameter is updated to indicate the value of the Reference to Data parameter when you save the metamap. You must correct the Reference to Data parameter if this value is invalid.</td>
</tr>
</tbody>
</table>

**User-Defined Properties**

You can optionally configure user-defined properties to add metadata not defined in the metamap. When you import the mapping specification, the Repository Service creates the property as a metadata extension.

The mapping specification templates do not contain a column to add user-defined properties. If you want to add the properties to a mapping specification template, modify the template to add a column that contains the metadata for the user-defined property.

The following table describes the parameters you can configure for user-defined properties:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name Style</td>
<td>Determines whether the name of the user-defined property occurs in more than one cell in the mapping specification template. Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Column. Metadata occurs more than once in a column of the mapping specification template. When you configure a reference to a column, the Repository Service processes each row in the column. The column can have empty rows.</td>
</tr>
<tr>
<td></td>
<td>- Cell. Metadata is in one cell in the mapping specification template.</td>
</tr>
<tr>
<td>Name Reference or Constant</td>
<td>Defines the name of the user-defined property. Contains a constant value or the location of a cell in the mapping specification template. To add the cell location, see “Modifying the Cell Location of Metadata” on page 46.</td>
</tr>
<tr>
<td>Value Style</td>
<td>Determines whether the value of the user-defined property occurs in more than one cell in the mapping specification template. Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Column. Metadata occurs more than once in a column of the mapping specification template. When you configure a reference to a column, the Repository Service processes each row in the column. The column can have empty rows.</td>
</tr>
<tr>
<td></td>
<td>- Cell. Metadata is in one cell in the mapping specification template.</td>
</tr>
<tr>
<td>Value Reference or Constant</td>
<td>Defines the value of the user-defined property. Contains a constant value or the location of a cell in the mapping specification template. To add the cell location, see “Modifying the Cell Location of Metadata” on page 46.</td>
</tr>
</tbody>
</table>
Modifying the Cell Location of Metadata

The Excel Table Parameters, Required Properties, and User-Defined Properties sections in the metamap worksheets contain the cell locations of metadata in a mapping specification template. For example, if you define a source table in a mapping specification template, you provide the table properties such as the columns in the table, the table business name, and the table description. The metamap references where each table property is located in the mapping specification template. The metamap stores each cell location on the SourceClasses, SourceAttributes, and SourceDatatypes worksheets.

Each reference to metadata in a mapping specification template contains the following information:

- **Mapping specification template name.** Name of the mapping specification template that corresponds to the metamap.
- **Worksheet name.** Name of the worksheet in the mapping specification template.
- **Cell location.** Row and column in the worksheet that contains the metadata.

When you select a reference cell in the metamap, the function bar in Excel contains the mapping specification template name, the worksheet name, and the cell location.
The following figure shows the Required Properties section of a SourceClasses metamap worksheet with the Class Name Reference to Data cell selected:

The function bar shows the following value of the Class Name Reference to Data cell:

=\[\text{MappingTemplate.xls}\]Source!$C$3

The following figure shows the corresponding section of the mapping specification template:

Cell C3 contains the Business Name metadata.

**Steps to Modify a Cell Location in the Metamap**

To modify a cell location in the metamap, copy and paste the cell location of a property from one of the following file types to the metamap:

- **Mapping specification template.** Use if your custom metamap is based on the Mapping metamap.
- **Repeating sheet file.** Use if your custom metamap is based on the Source-Target-Matrix metamap.

**To modify a cell location in the metamap:**

1. Open the mapping specification template or repeating sheet file and the metamap.
2. In the mapping specification template or repeating sheet file, navigate to the cell for the metadata that you need to reference in the metamap.
3. Right-click the cell and select Copy.
4. In the metamap, navigate to the appropriate worksheet.
5. Right-click one of the following cells and click Paste Special:
   - Detection Parameter cell in the Excel Table Parameters section
   - Reference to Data cell for the appropriate Metadata Field in the Required Properties section
6. Click Paste Link.

The metapmap stores the location of the property in the mapping specification template or repeating sheet file. The location includes the name of the mapping specification template or repeating sheet file and the worksheet name that contains the cell.

## Import and Export Parameters in the Metamap

The Overview worksheet in the metamap contains default import and export parameters. The Repository Service reads the import parameters when you import a mapping specification. The Repository Service reads the export parameters when you export a mapping specification. You can optionally override the default parameters by editing the values in the Excel metamap and saving it as an XML file.

### Import Parameters

The following table describes the parameters you configure to import a mapping specification into the PowerCenter repository:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compare names ignoring case</td>
<td>Determines whether name evaluation is case sensitive. Select one of the following options: - True. Ignore case when evaluating names. - False. Performs a case-sensitive evaluation. Default is True.</td>
</tr>
<tr>
<td>Merge objects with same name</td>
<td>Merge objects with the same name in the mapping specification. Select one of the following options: - True. Merge objects with the same name. - False. Keep all objects. Use this option when the mapping specification includes different objects with the same name. Default is True.</td>
</tr>
<tr>
<td>ETL data stream merging strategy</td>
<td>Action to take when the mapping specification contains two or more sources without a join expression. Select one of the following options: - Add joins if needed. Adds a Joiner transformation to the mapping. - Add union if needed. Use separate pipelines for sources. - Do not change. Do not add a join or a union. Default is Add joins if needed.</td>
</tr>
<tr>
<td>Print warnings if rows are skipped</td>
<td>Print warnings in the Import Metadata wizard when rows are skipped. The Import Wizard displays errors when you import mapping specifications. Select one of the following options: - True. Print warnings for skipped rows. - False. Do not print warnings for skipped rows. Default is False.</td>
</tr>
</tbody>
</table>
Export Parameters

The following table describes the parameters you configure to export a mapping from the PowerCenter repository to a mapping specification:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output package scope</td>
<td>Type of metadata to export. Select one of the following options: - All. Export all metadata. - Mapping sources only. Export source definitions. - Mapping targets only. Export target definitions. - Mapping sources and targets. Export source and target definitions. Default is All. Select the same option for Input package scope and Output package scope.</td>
</tr>
<tr>
<td>Input package scope</td>
<td>Type of metadata to export. Select one of the following options: - All. Export all metadata. - Mapping sources only. Export source definitions. - Mapping targets only. Export target definitions. - Mapping sources and targets. Export source and target definitions. Default is All. Select the same option for Input package scope and Output package scope.</td>
</tr>
<tr>
<td>Input design level scope</td>
<td>PowerCenter always exports metadata at a physical design level.</td>
</tr>
<tr>
<td>Organize cells which have the same value</td>
<td>Display options for duplicate values in adjacent cells in the mapping specification. Select one of the following options: - Merge cells having the same value. Merge cells that would contain the same value to reduce duplication of information. - Put value only in the first cell. Write the value in the first applicable cell. - Put value into all cells. Repeat values in all applicable cells. Default is Merge cells.</td>
</tr>
<tr>
<td>Draw borders</td>
<td>Draws the bottom rule for each row in the mapping specification. Select one of the following options: - True. Draws a bottom rule for each row. - False. Does not draw rules. Default is True.</td>
</tr>
<tr>
<td>Output classes not used in mappings</td>
<td>Export source or target definitions that appear in the mapping but are not used. Applies when the template type option in the metamap is configured for mappings. Select one of the following options: - True. Exports all supported information. - False. Does not export information that is not used in the mapping. Default is True.</td>
</tr>
<tr>
<td>Share common data types</td>
<td>Datatype definitions in the mapping specification. Use this option to override datatype definition in the exported mapping specification. Select one of the following options: - True. Use a single datatype definition for the file. Use when you define each datatype in the mapping specification and want all columns that use a datatype to use the same definition. - False. Use different definitions for the same datatype. Use when you want to allow a datatype to be defined differently within the mapping specification. Default is True.</td>
</tr>
</tbody>
</table>

Saving the Metamap as an XML Spreadsheet

When you finish editing the metamap, save the metamap as an XML spreadsheet. The Mapping Analyst for Excel import and export process uses the XML version of the metamap.
To save the metatmap as an XML spreadsheet:

1. In the metatmap, click File > Save As.
2. For Save As Type, select XML Spreadsheet.
3. Click Save.
Customizing Mapping Specification Templates - Example

This chapter includes the following topics:
♦ Overview, 51
♦ Step 1. Copy the Mapping Specification Template, Repeating Sheet File, and Metamap, 52
♦ Step 2. Associate the Metamap with the Repeating Sheet File, 52
♦ Step 3. Edit the Mapping Specification Template and the Repeating Sheet File, 53
♦ Step 4. Edit the Metamap, 53

Overview

As a business analyst, you want to create multiple mappings in a single mapping specification. You can use the Source-Target-Matrix template to create multiple mappings on separate worksheets. When you import mapping specifications based on the Source-Target-Matrix template, the Repository Service uses the worksheet name for both the mapping and target definition names. However, you want to use the names of the target database tables for the target definition names in the mapping.

The Source-Target-Matrix template does not include a target name column. To add a target name column to the template, configure a custom mapping specification template based on the Source-Target-Matrix template.

To customize the Source-Target-Matrix mapping specification template, copy and modify the following files:
♦ Source-Target-Matrix mapping specification template (source-target-matrix.xls). Includes a separate worksheet for each mapping in the mapping specification.
♦ Source-Target-Matrix repeating sheet file (source-target-matrix-mirrepeatingssh.e.xls). Defines the common structure for multiple mappings defined in a mapping specification that is based on the Source-Target-Matrix template. When you modify the Source-Target-Matrix template, you must make the same changes to a custom version of the Source-Target-Matrix repeating sheet file.

To customize the Source-Target-Matrix mapping specification template, complete the following steps:
1. Copy the mapping specification template, repeating sheet file, and the metamap.
2. Associate the metamap with the repeating sheet file.
3. Edit the mapping specification template and the repeating sheet file.
4. Edit the metamap to read the repeating sheet file.

Step 1. Copy the Mapping Specification Template, Repeating Sheet File, and Metamap

To avoid altering the original files, copy and rename the Source-Target-Matrix template, the repeating sheet file, and the associated metamap.

A metamap uses relative links to find the location of metadata in a mapping specification. To avoid broken links, use Windows Explorer to copy the templates. Do not use the Save As menu option in Microsoft Excel.

The original files are located in the following directory:

`<PowerCenter Client installation directory>\bin\Sample Specifications`

Copy and rename the following files:

- `source-target-matrix.xls`
- `source-target-matrix-mirrepeatingsheet.xls`
- `source-target-matrix-metamap.xls`

For example, you can give the copied files the following names:

- `custom-source-target-matrix.xls`
- `custom-stm-mirrepeatingsheet.xls`
- `custom-stm-metamap.xls`

Step 2. Associate the Metamap with the Repeating Sheet File

The Source-Target-Matrix metamap reads from the repeating sheet file to determine the structure of the mapping on each worksheet of a mapping specification that is based on the Source-Target-Matrix template. When you rename the repeating sheet file, you must associate the metamap with the new repeating sheet file name to maintain the metamap links to the repeating sheet file.

**To associate the metamap with the repeating sheet file:**

1. Open the new repeating sheet file.
   For example, open the following file:
   `custom-stm-mirrepeatingsheet.xls`

2. Open the new metamap.
   For example, open the following file:
   `custom-stm-metamap.xls`
   If a security warning appears to warn against using macros, click Enable Macros to enable the validation macros.

3. When a message box states that the metamap contains links to other data sources, click Update.
   A message states that one or more links cannot be updated.

4. Click Edit Links.
   The Edit Links dialog box appears.
5. Click Change Source.
6. Select the new repeating sheet file, and click OK.
7. Click Close to close the Edit Links dialog box.
8. Save the metemap.

Step 3. Edit the Mapping Specification Template and the Repeating Sheet File

Add a target name column to the new mapping specification template and repeating sheet file so that you can configure names for imported target definitions.

The mapping specification template and repeating sheet file contain several columns grouped together under a Target heading. To avoid additional format changes, add a column to the right of the first target column, and then name the columns.

To add a target name column:

1. Open the repeating sheet file.
   
   For example, open the following file:
   
   custom-stm-mirrepeatingssheet.xls

2. On the _MIR_REPEATING_SHEET_ worksheet, click in the Target Field/Column Datatype Native Name column.

3. Click Insert > Column.
   
   A new column displays on the left side of the column.

4. In the new column, enter Field/Column for the column name.

5. Rename the original Field/Column name to File/Table.
   
   This is the new target name column.

6. Save the changes.

7. Open the new mapping specification template.
   
   For example, open the following file:
   
   custom-source-target-matrix.xls

8. Repeat steps 2 through 6 to add a target name column to the following worksheets:
   
   ♦ mapping1
   ♦ mapping2
   ♦ mapping3

Step 4. Edit the Metamap

Edit the metemap to describe the location of metadata in the new repeating sheet file. When you added the target name column to the Source-Target-Matrix mapping specification template and repeating sheet file, the location of all subsequent columns changed.
For example, in the original Source-Target-Matrix mapping specification template and repeating sheet file, target field and column data is located in columns F through L. In the modified Source-Target-Matrix mapping specification template and repeating sheet file, column F contains target table and file information, and field and column data begins in column G. The location of all remaining columns changed.

Complete the following tasks to edit the metamap to reflect the changes to the target metadata:

1. Edit the TargetClasses worksheet.
2. Edit the TargetAttributes worksheet.
3. Edit the FeatureMaps worksheet.
4. Save the metamap as an XML spreadsheet.

**Edit the TargetClasses Worksheet**

The TargetClasses worksheet in the metamap defines the metadata location of the name of target files or relational tables.

**To edit the TargetClasses worksheet:**

1. Open the new repeating sheet file and metamap.
   
   For example, open the following files:
   
   - custom-stm-mirrepeatingsheet.xls
   - custom-stm-metamap.xls

2. In the repeating sheet file, navigate to the first cell in the new Target File/Table column, cell F3.

3. Right-click the cell and select Copy.

4. In the metamap, navigate to the TargetClasses worksheet.

5. In the Excel Table Parameters section, right-click the First Line Detection Parameter cell and click Paste > Special.

6. Click Paste Link.

   Microsoft Excel pastes the new location to start reading for a target table or file name. The cell location displays in the function bar. In this example, the new location is cell F3.

   The following figure shows the modified First Line Detection Parameter cell on the TargetClasses worksheet:

7. Set the Last Line How to Detect cell to Count of Empty Lines.

8. Set the Last Line Detection Parameter cell to 100.

   With this setting, the Repository Service allows 100 blank lines in the target File/Table column before it stops reading for another target table or file name. This allows 100 column or field names for each target. If you require more columns or fields for targets, increase the Last Line Detection Parameter.

9. In the Required Properties section, set the Class Name Style cell to Column.
10. In the repeating sheet file, navigate to the first cell in the new Target File/Table column, cell F3.

11. Right-click the cell and select Copy.

12. In the metamap, navigate to the TargetClasses worksheet.

13. In the Class Name Reference to Data cell, right-click and choose Paste Special.

14. Click Paste Link.

The following figure shows the modified Class Name metadata field on the TargetClasses worksheet:

![Class Name Style and Reference to Data cells]

15. Save the changes.

**Edit the TargetAttributes Worksheet**

The TargetAttributes worksheet in the metamap defines the metadata location of fields or columns in a target file or relational table.

**To edit the TargetAttributes worksheet:**

1. In the repeating sheet file, navigate to the first cell in the new Target File/Table column, cell F3.

2. Right-click the cell and select Copy.

3. In the metamap, navigate to the TargetAttributes worksheet.

4. On the TargetAttributes worksheet, set the Class Name Style cell to Column.

5. In the Class Name Reference to Data cell, right-click and choose Paste Special.

6. Click Paste Link.

Microsoft Excel pastes the new location to start reading for a target table or file name. The cell location displays in the function bar. In this example, the new location is cell F3.
The following figure shows the modified Class Name Reference to Data cell on the TargetAttributes worksheet:

<table>
<thead>
<tr>
<th>Target Column in Repeating Sheet File</th>
<th>Copy Start Cell</th>
<th>Metamap Worksheet</th>
<th>Cell to Paste as Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field/Column</td>
<td>G3</td>
<td>TargetAttributes</td>
<td>Attribute Name Reference to Data cell</td>
</tr>
<tr>
<td>Field/Column Datatype Native Name</td>
<td>H3</td>
<td>TargetAttributes</td>
<td>Datatype Native Name Reference to Data cell</td>
</tr>
<tr>
<td>Field/Column Datatype</td>
<td>I3</td>
<td>TargetAttributes</td>
<td>Datatype Name Reference to Data cell</td>
</tr>
<tr>
<td>Length</td>
<td>J3</td>
<td>TargetAttributes</td>
<td>Datatype Length Reference to Data cell</td>
</tr>
<tr>
<td>Scale</td>
<td>K3</td>
<td>TargetAttributes</td>
<td>Datatype Scale Reference to Data cell</td>
</tr>
<tr>
<td>Description</td>
<td>L3</td>
<td>TargetAttributes</td>
<td>Attribute Description Reference to Data cell</td>
</tr>
<tr>
<td>Notes</td>
<td>M3</td>
<td>TargetAttributes</td>
<td>Attribute Comment Reference to Data cell</td>
</tr>
</tbody>
</table>

7. Repeat steps 1 through 6 for each target column in the repeating sheet file as described in the following table:

8. Save the changes.

**Edit the FeatureMaps Worksheet**

The FeatureMaps worksheet in the metamap defines a mapping between a source and target.

**To edit the FeatureMaps worksheet:**

1. In the repeating sheet file, navigate to the first cell in the new Target File/Table column, cell F3.
2. Right-click the cell and select Copy.
3. In the metamap, navigate to the FeatureMaps worksheet.
4. In the Feature Map Target Class Name Reference to Data cell, right-click and choose Paste Special.
5. Click Paste Link.

Microsoft Excel pastes the new location to start reading for a target table or file name. The cell location displays in the function bar. In this example, the new location is cell F3.
6. In the repeating sheet file, navigate to the first cell in the new Target Field/Column column, cell G3.
7. Right-click the cell and select Copy.
8. In the metamap, navigate to the FeatureMaps worksheet.
9. In the Feature Map Target Attribute Name Reference to Data cell, right-click and choose Paste Special.
10. Click Paste Link.

Microsoft Excel pastes the new location to start reading for a target column or field name. The cell location displays in the function bar. In this example, the new location is cell G3.

The following figure shows the modified Feature Map Target Attribute Name Reference to Data cell on the FeatureMaps worksheet:

11. Save the changes.

Save the Metamap as an XML Spreadsheet

When you finish editing the metamap, save the metamap as an XML spreadsheet. You use an XML version of the metamap when you import mapping specifications or export mappings.

To save the metamap as an XML spreadsheet:
1. In the metamap, click File > Save As.
2. For Save As Type, select XML Spreadsheet.
3. Click Save.

You can now use the new mapping specification template and metamap to create and import mappings that include user-defined names for target definitions.
Metamap Worksheets

This appendix includes the following topics:

- Overview Worksheet, 59
- Models Worksheet, 59
- Source and Target Worksheets, 60
- Mapping Worksheets, 62
- MITI_SystemTypes Worksheet, 64
- MITI_DataTypes Worksheet, 64

Overview Worksheet

The Overview worksheet in a metamap contains options that determine how the Repository Service imports and exports metadata and the template type you want to use.

You can configure the import and export parameters. The import parameters determine how the Repository Service imports the metadata in the mapping specification. The export parameters determine how the Repository Service exports metadata from the repository to a mapping specification.

Select one of the following template types from the Overview worksheet:

- **Mapping.** Select to configure sources, targets, and mappings in a mapping specification. When selected, the metamap displays all worksheets.
- **Data Dictionary.** Select to configure source tables or files in a mapping specification. When selected, the metamap displays source worksheets only.

The default template type is Mapping.

Models Worksheet

The Models worksheet in a metamap contains references to the entity that contains all metadata. The metadata fields in the Required Properties section contain the metadata references. The Mapping metamap and the Source-Target-Matrix metamap do not use all of the metadata fields in the Models worksheet.
The following table describes the metadata fields in the Models worksheet that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Name</td>
<td>Mapping Source-Target-Matrix</td>
<td>Name of the model. Required by the Mapping Analyst for Excel import process. The Repository Service does not import this value.</td>
</tr>
</tbody>
</table>

Source and Target Worksheets

The source and target worksheets in a metmap contain references to source and target metadata. The Repository Service reads each worksheet in the hierarchy to create a source or target definition.

The metadata fields in the Required Properties section contain the metadata references. The Mapping metamap and the Source-Target-Matrix metamap do not use all of the metadata fields in the source and target worksheets.

DataPackages Worksheets

The SourceDataPackages and TargetDataPackages worksheets contain references to the type of the source or target database or file system, such as Oracle, DB2, or flat file.

The following table describes the metadata fields in the SourceDataPackages and TargetDataPackages worksheets that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Name</td>
<td>Mapping Source-Target-Matrix</td>
<td>Name of the data package. Required by the Mapping Analyst for Excel import process. The Repository Service does not import this value.</td>
</tr>
<tr>
<td>Package Description</td>
<td>Mapping</td>
<td>Description of the data package. The Repository Service does not import this value.</td>
</tr>
<tr>
<td>System Type</td>
<td>Mapping Source-Target-Matrix</td>
<td>Type of the source or target database or file system.</td>
</tr>
</tbody>
</table>

Schemas Worksheets

The SourceSchemas and TargetSchemas worksheets contain references to the name of the source or target database or file system.

The following table describes the metadata fields in the SourceSchemas and TargetSchemas worksheets that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Name</td>
<td>Mapping Source-Target-Matrix</td>
<td>Identifies the parent data package.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Mapping Source-Target-Matrix</td>
<td>Name of the source or target database or file system.</td>
</tr>
<tr>
<td>Schema Comment</td>
<td>Mapping Source-Target-Matrix</td>
<td>Description of the source or target database or file system. The Repository Service does not import this value.</td>
</tr>
</tbody>
</table>

Classes Worksheets

The SourceClasses and TargetClasses worksheets contain references to source and target tables and files.
The following table describes the metadata fields in the SourceClasses and TargetClasses worksheets that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Name</td>
<td>Mapping</td>
<td>Identifies the parent data package.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Mapping</td>
<td>Identifies the parent schema.</td>
</tr>
<tr>
<td>Class Name</td>
<td>Mapping</td>
<td>Business name of the source or target definition.</td>
</tr>
<tr>
<td>Class Physical Name</td>
<td>Mapping</td>
<td>Name of the source or target definition.</td>
</tr>
<tr>
<td>Class Description</td>
<td>Mapping</td>
<td>Description of the source or target definition.</td>
</tr>
</tbody>
</table>

**Attributes Worksheets**

The SourceAttributes and TargetAttributes worksheets contain references to fields or columns in a relational table or file.

The following table describes the metadata fields in the SourceAttributes and TargetAttributes worksheets that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Name</td>
<td>Mapping</td>
<td>Identifies the parent data package.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Mapping</td>
<td>Identifies the parent schema.</td>
</tr>
<tr>
<td>Class Name</td>
<td>Source-Target-Matrix</td>
<td>Identifies the parent class.</td>
</tr>
<tr>
<td>Class Physical Name</td>
<td>Mapping</td>
<td>Identifies the parent class.</td>
</tr>
<tr>
<td>Attribute Name</td>
<td>Mapping</td>
<td>Business name for the column.</td>
</tr>
<tr>
<td>Attribute Physical Name</td>
<td>Mapping</td>
<td>Column name.</td>
</tr>
<tr>
<td>Attribute Description</td>
<td>Mapping</td>
<td>Description for the column.</td>
</tr>
<tr>
<td>Attribute Comment</td>
<td>Source-Target-Matrix</td>
<td>Description for the column. When used, overrides the Attribute Description metadata field. Used on the TargetAttributes worksheet only.</td>
</tr>
<tr>
<td>Data Type Name</td>
<td>Source-Target-Matrix</td>
<td>Datatype of the column.</td>
</tr>
<tr>
<td>Data Type</td>
<td>Mapping</td>
<td>Datatype of the column.</td>
</tr>
<tr>
<td>Data Type Native Name</td>
<td>Source-Target-Matrix</td>
<td>Datatype of the column. When used, overrides the Data Type Name metadata field. Used on the TargetAttributes worksheet only.</td>
</tr>
<tr>
<td>Data Type Length</td>
<td>Mapping</td>
<td>Precision of the column.</td>
</tr>
<tr>
<td>Data Type Scale</td>
<td>Mapping</td>
<td>Scale of the column.</td>
</tr>
</tbody>
</table>

**DataTypes Worksheets**

The SourceDataTypes and TargetDataTypes worksheets contain references to datatype definitions. Configure the datatypes worksheets when you define enumerated data types in a mapping specification template.
The Mapping and Source-Target-Matrix mapping specification templates do not include enumerated data types. As a result, the associated metatags do not use the datatypes worksheets.

Mapping Worksheets

The mapping worksheets in a metamap contain references to the location of mapping and transformation metadata, including how to connect the source and target columns. The Repository Service reads each worksheet in the hierarchy to create a mapping.

The metadata fields in the Required Properties section contain the metadata references. The Mapping metamap and the Source-Target-Matrix metamap do not use all of the metadata fields in the mapping worksheets.

MappingDesignPackages Worksheet

The MappingDesignPackages worksheet contains references to the entity that contains all mapping metadata. The following table describes the metadata field in the MappingDesignPackages worksheet that the Informatica metatags use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Name</td>
<td>Mapping</td>
<td>Name of the mapping design package. Required by the Mapping Analyst for Excel import process. The Repository Service does not import this value.</td>
</tr>
<tr>
<td>Transformation Task Name</td>
<td>Mapping</td>
<td>Identifies the parent mapping design package.</td>
</tr>
<tr>
<td>Transformation Task Name</td>
<td>Source-Target-Matrix</td>
<td>Mapping name.</td>
</tr>
</tbody>
</table>

TransformationTasks Worksheet

The TransformationTasks worksheet contains references to the mapping name. The following table describes the metadata fields in the TransformationTasks worksheet that the Informatica metatags use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Name</td>
<td>Mapping</td>
<td>Identifies the parent mapping design package.</td>
</tr>
<tr>
<td>Transformation Source Schema Name</td>
<td>Mapping</td>
<td>Name of the source database or file system to filter.</td>
</tr>
<tr>
<td>Transformation Source Class Physical Name</td>
<td>Mapping</td>
<td>Name of the source table to filter.</td>
</tr>
</tbody>
</table>

Filters Worksheet

The Filters worksheet contains references to expressions to filter source data. The following table describes the metadata fields in the Filters worksheet that the Informatica metatags use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter Condition</td>
<td>Mapping</td>
<td>Filter condition.</td>
</tr>
<tr>
<td>Transformation Source Schema Name</td>
<td>Mapping</td>
<td>Name of the source database or file system to filter.</td>
</tr>
<tr>
<td>Transformation Source Class Physical Name</td>
<td>Mapping</td>
<td>Name of the source table to filter.</td>
</tr>
</tbody>
</table>
Lookups Worksheet

The Lookups worksheet contains references to expressions that return data from a lookup source.

The following table describes the metadata fields in the Lookups worksheet that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lookup Condition</td>
<td>Mapping</td>
<td>Lookup condition.</td>
</tr>
<tr>
<td>Transformation Source Schema Name</td>
<td>Mapping</td>
<td>Name of the source database or file system to look up.</td>
</tr>
<tr>
<td>Transformation Source Class Physical Name</td>
<td>Mapping</td>
<td>Name of the source table to look up.</td>
</tr>
</tbody>
</table>

Joins Worksheet

The Joins worksheet contains references to metadata that joins two source definitions.

The following table describes the metadata fields in the Joins worksheet that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Join Type</td>
<td>Mapping</td>
<td>Type of join.</td>
</tr>
<tr>
<td>Left Attribute Schema Name</td>
<td>Mapping</td>
<td>Name of the source database or file system containing the tables to join.</td>
</tr>
<tr>
<td>Left Attribute Class Physical Name</td>
<td>Mapping</td>
<td>Name of the left or master table.</td>
</tr>
<tr>
<td>Left Attribute Physical Name</td>
<td>Mapping</td>
<td>Name of the left or master column.</td>
</tr>
<tr>
<td>Right Attribute Schema Name</td>
<td>Mapping</td>
<td>Name of the source database or file system containing the tables to join.</td>
</tr>
<tr>
<td>Right Attribute Class Physical Name</td>
<td>Mapping</td>
<td>Name of the right or detail table.</td>
</tr>
<tr>
<td>Right Attribute Physical Name</td>
<td>Mapping</td>
<td>Name of the right or detail column.</td>
</tr>
</tbody>
</table>

FeatureMaps Worksheet

The FeatureMaps worksheet contains references to the following mapping operations:

- Connections between source and target ports
- Aggregate and non-aggregate expressions

The following table describes the metadata fields in the FeatureMaps worksheet that the Informatica metamaps use:

<table>
<thead>
<tr>
<th>Metadata Field</th>
<th>Used by Metamap</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transformation Task Name</td>
<td>Source-Target-Matrix</td>
<td>Identifies the parent transformation task.</td>
</tr>
<tr>
<td>Feature Map Operation</td>
<td>Mapping</td>
<td>Connections between source and target ports. This can include an aggregate or non-aggregate expression.</td>
</tr>
<tr>
<td>Feature Map Operation Description</td>
<td>Mapping</td>
<td>Description for an aggregate or non-aggregate expression.</td>
</tr>
</tbody>
</table>
### MITI_SystemTypes Worksheet

The MITI_SystemTypes worksheet lists the file and relational source and target types that Mapping Analyst for Excel supports.

### MITI_DataTypes Worksheet

The MITI_DataTypes worksheet lists native datatypes and the corresponding Mapping Analyst for Excel datatypes.

For more information, see "Mapping Analyst for Excel Datatypes" on page 8.
This appendix includes the following topic:

- Mapping Analyst for Excel Glossary of Terms, 65

Mapping Analyst for Excel Glossary of Terms

**mapping specification**
A Microsoft Excel file that contains mapping metadata to import into the PowerCenter repository. The mapping metadata can include sources, targets, and transformations. Each mapping specification requires an associated metamap.

**mapping specification template**
Microsoft Excel files installed with the PowerCenter Client that define the structure for a mapping specification. Use a mapping specification template as a starting point to create a mapping specification.

**metamap**
A Microsoft Excel or XML file that contains a hierarchical representation of the metadata in a mapping specification. The metamap contains import and export options and cell references to metadata in a mapping specification.

**repeating sheet file**
A Microsoft Excel file that defines the common structure for multiple mappings defined in a mapping specification that is based on the Source-Target-Matrix mapping specification template.
INDEX

A
Aggregator transformation
configuring in mapping specification 11

C
custom
mapping specification template 41
metamap 39
repeating sheet file 41

D
datatypes
defining in mapping specification 8, 24, 25
defining in metamap 61

E
Excel table parameters
metamap 43
export parameters
metamap 49
exporting
to mapping specifications 36
Expression transformation
configuring in mapping specification 12, 25

F
FeatureMaps worksheet
metamap 63
Filter transformation
configuring in mapping specification 11
Filters worksheet
metamap 62
functions
adding to mapping specification 9
Functions worksheet
Mapping template 9

I
import parameters
metamap 48

importing
from Microsoft Excel, process 2
mapping specifications 34

J
Joiner transformation
configuring in mapping specification 9
Joins worksheet
metamap 63

L
Lookup transformation
configuring in mapping specification 10
Lookups worksheet
metamap 63

M
Main worksheet
Mapping template 6
Mapping Analyst for Excel
components 1
datatypes 8
element multiple mappings 27
element one mapping 15
metamap hierarchy 39
overview 1
process 2
templates 3
mapping specification
configuring multiple mappings 23
configuring single mapping 5
creating custom templates 41
custom example 51
defining datatypes 8, 24, 25
definition 1
exporting to 36
importing 34
multiple mappings example 27
one mapping example 15
mapping specification templates
custom 39, 41
definition 3
element 15, 27
using 5, 27
Mapping template
  adding functions 9
  Aggregator transformation 11
  configuring transformations 9
description 3
  example 15
  Expression transformation 12
  Filter transformation 11
  Functions worksheet 9
  Joiner transformation 9
  joining source data 9
  Lookup transformation 10
  Main worksheet 6
  Source worksheet 7
Target worksheet 7
  using 5
  validating 12
mapping worksheets
  metamap 62
MappingDesignPackages worksheet
  metamap 62
metadata
  metamap hierarchy 39
  referencing in metamap 46
metamap
  associating with template 42
  creating custom 39
  custom example 51
  defining datatypes 61
  definition 2
  editing 43
  editing example 53
  Excel table parameters 43
  export parameters 49
  FeatureMaps worksheet 63
  Filters worksheet 62
  hierarchy 39
  import parameters 48
  Joins worksheet 63
  Lookups worksheet 63
  mapping worksheets 62
  MappingDesignPackages worksheet 62
  metadata hierarchy 39
  MITI_DataTypes worksheet 64
  MITI_SystemTypes worksheet 64
Models worksheet 59
  Overview worksheet 59
  referencing metadata 44, 46
required properties 44
  source worksheets 60
SourceAttributes worksheet 61
  SourceClasses worksheet 60
SourceDataPackages worksheet 60
SourceDataTypes worksheet 61
SourceSchemas worksheet 60
target worksheets 60
TargetAttributes worksheet 61
  TargetClasses worksheet 60
  TargetDataPackages worksheet 60
  TargetDataTypes worksheet 61
  TargetSchemas worksheet 60
TransformationTasks worksheet 62
user-defined properties 45

XML version 49
Microsoft Excel
  creating PowerCenter metadata 1
  exporting to 36
  importing from 34
MITI_DataTypes worksheet
  metamap 64
MITI_SystemTypes worksheet
  metamap 64
Models worksheet
  metamap 59
multiple mappings
  creating a single mapping specification 23
  multiple sources
    configuring in mapping specification 9

O
  Overview worksheet
    metamap 59

R
  Reference to Data
    metamap column 44
  repeating sheet file
    custom 41
    description 3
    editing example 53
required properties
    metamap 44

S
  Source worksheet
    Mapping template 7
    source worksheets
      metamap 60
SourceAttributes worksheet
  metamap 61
SourceClasses worksheet
  metamap 60
SourceDataPackages worksheet
  metamap 60
SourceDataTypes worksheet
  metamap 61
SourceSchemas worksheet
  metamap 60
Source-Target-Matrix template
  customizing 51
description 3
  example 27
  Expression transformation 25
  sources
    24
  targets 25
using 23
T

Target worksheet  
  Mapping template 7

Target worksheets  
  metamap 60

TargetAttributes worksheet  
  metamap 61

TargetClasses worksheet  
  metamap 60

TargetDataPackages worksheet  
  metamap 60

TargetDataTypes worksheet  
  metamap 61

targets  
  configuring in mapping specification 7, 25

TargetSchemas worksheet  
  metamap 60

templates  
  custom mapping specification 39

  mapping specification 3

transformations  
  configuring in mapping specification 9, 25

TransformationTasks worksheet  
  metamap 62

U

user-defined properties  
  metamap 45

V

validation  
  Mapping template 12

X

XML spreadsheet  
  metamap 49
This Informatica product (the “Software”) includes certain drivers (the “DataDirect Drivers”) from DataDirect Technologies, an operating company of Progress Software Corporation (“DataDirect”) which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.

2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.