## Contents

**Preface** ........................................................................................................................................................................... v

### 1 Introduction

Integration with Oracle Fusion Product Data Hub ........................................................................................................... 1-1
Fusion PIMDH Configuration ............................................................................................................................................... 1-2
ICC Structure as Product Hierarchy ............................................................................................................................... 1-2
Relationship Between ICC and Semantic Model ........................................................................................................... 1-3
Cleansing and Matching Item On-Boarding Process ....................................................................................................... 1-4

### 2 Implementing and Configuring

Getting Started ................................................................................................................................................................. 2-1
Preparing for Implementation ........................................................................................................................................ 2-1
EDQP for Fusion PIMDH Package Contents .................................................................................................................. 2-2
Implementing the EDQP for Fusion PIMDH Files ........................................................................................................... 2-2
Establishing the EDQP Oracle DataLens Server Connection ......................................................................................... 2-3
Enabling Fusion PIMDH DQ Processing ........................................................................................................................ 2-3

### 3 Using the Fusion PIMDH DQ Synchronization Process

Metadata and Data Lens Synchronization Process ........................................................................................................ 3-1
Exporting ICC Metadata from Fusion PIMDH ................................................................................................................ 3-2
Item Class & Attribute Metadata Export ...................................................................................................................... 3-2
Alternate Catalog Export ............................................................................................................................................... 3-7
Item Class Sample Item Export ................................................................................................................................... 3-8
Importing a Metadata File into Excel ........................................................................................................................ 3-8
Creating a Data Lens ....................................................................................................................................................... 3-8
Updating an Existing Data Lens ................................................................................................................................... 3-10
Adding an Alternate Catalog to a Data Lens .................................................................................................................. 3-11
Changing the Data Lens in the DQ Processing DSA ...................................................................................................... 3-12
Refreshing Metadata and Semantic Keys ....................................................................................................................... 3-13

### 4 Continued DQ Improvement

Refining Your New Data Lens ......................................................................................................................................... 4-1
Using Value Sets when Creating a Data Lens .................................................................................................................. 4-4
Refining Value Sets in a Data Lens .................................................................................................................................. 4-6
Updating Your Fusion PIMDH DQ Metadata Structure ................................................................................................. 4-7
Preface

This manual highlights the implementation and configuration of Oracle Enterprise Data Quality for Product Data to work with Fusion PIMDH. Additionally, it describes how to synchronize the data in Fusion PIMDH with your Enterprise DQ for Product data lenses.

By extending Enterprise DQ for Product capabilities, Oracle enables customers to improve the accuracy, completeness, and integrity of their master product data.

To understand the concepts presented, you must use this reference guide in conjunction with the Enterprise DQ for Product documents listed in "Related Documents" on page 2-v.

Audience

You should have a basic understanding of the DataLens Technology. This includes the functionality of the Oracle Enterprise Data Quality for Product Data Knowledge Studio and Application Studio applications.

This document is intended for all users of the DataLens Technology, including:
- Business Users that implement and configure the product and refine data lenses.
- IT Administrators that assist in implementing and configuring the product.

Documentation Accessibility

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

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For information about configuring Fusion PIM Data Hub to use the Data Quality capabilities, see the Item Data Quality Management white paper at My Oracle Support: https://support.us.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx?id=1394702.1
For more information, see the following documents in the documentation set:

- The *Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Installation Guide* provides detailed Oracle DataLens Server installation instructions.
- The *Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Administration Guide* provides information about installing and managing an Oracle DataLens Server.
- The *Oracle Enterprise Data Quality for Product Data AutoBuild Reference Guide* provides information about creating an initial data lens based on existing product information and data lens knowledge.
- The *Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide* provides information about creating and maintaining data lenses.
- The *Oracle Enterprise Data Quality for Product Data Services for Excel Reference Guide* provides information about creating a DSA based on data contained in a Microsoft Excel worksheet.

See the latest version of this and all documents in the Oracle Enterprise Data Quality for Product Data Documentation Web site at http://docs.oracle.com/cd/E35636_01/index.htm

### Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
The Oracle DataLens Server can be configured to run with multiple servers:

- Oracle DataLens Administration Server
- Oracle DataLens Transform Server

The administration of all servers in a multi-server configuration is controlled with the Oracle DataLens Administration Server. The purpose of the Administration Server is to manage the various administrative tasks of the servers for the server groups (referred to as Transform Servers) and can itself serve as its own Transform Server when installed alone in a single node configuration. By spreading the data processing load across multiple servers the Oracle DataLens Server system provides scalability and configuration control over the various functional areas involved in developing, testing, and ultimately executing Oracle DataLens jobs.

The type of Oracle DataLens Server Group that a Transform Server belongs to controls the individual server functionality. A server group may contain one or many physical servers. There are three types of server groups:

- Development Server Group
- Production Server Group
- Quality Assurance Server Group

The Server Groups contain individual Oracle DataLens Servers on physical machines that can load balance among servers within the group. The data lenses and DataLens Service Applications (DSAs) are deployed from one group to the next beginning with the development group, then migrating to the Quality Assurance Group for testing before promotion to the Production Group for deployment to production. This multiple group migration function facilitates an enterprise business process where multiple functional areas work on data lens objects in stages before releasing them to production.

The integration of Oracle Fusion Product Information Management (PIM) and Enterprise DQ for Product (EDQP) adds a package called the "Fusion PIM" to the Enterprise DQ for Product solution upload package.

**Integration with Oracle Fusion Product Data Hub**

The integration of the EDQP solution with the Fusion PIM Data Hub (PIMDH) provides an integrated set of capabilities to categorize, standardize, match, govern, validate, and correct product data being introduced from any source system(s) or catalog(s). The EDQP Fusion PIMDH solution delivers an ready-to-use integration DSA that facilitates the real time data quality processes using the Fusion PIMDH Batch Import and the New Item Create processes.
**Fusion PIMDH Configuration**

The system can be configured to perform any or all of these checks on all batches of data being imported into the system using the Fusion PIMDH Import Workbench process. These configuration steps are fully described in this document. The integration is delivered with a set of preconfigured integration files that can be personalized to meet your project requirements. The files allow you to run operations such as populating user defined attributes (Form, Fit or Function) for an item on-boarding process using the EDQP system.

**ICC Structure as Product Hierarchy**

For most effective metadata synchronization, the PIM ICC structure should be set up as a product hierarchy. For example, you might have a four-level hierarchy consisting of the following levels:

L1: Category family
L2: Category group
L3: Category line
L4: Category

A product hierarchy contains multiple levels and each level is identified by a set of defining attributes. The levels must be non-overlapping and unique. Each item in PIM should belong to one and only one ICC. The category-specific descriptive attributes must be defined at the leaf-node level while the more general category attributes may be defined at higher ICC levels.

The following is an example of well-formed four-level product hierarchy:

![product hierarchy diagram]

In contrast, the following diagram shows an example of a business-unit/product hierarchy. This type of hierarchy combines business unit structure with product structure. Although this type of structure may be useful within business organizations, it can undermine the value of a PIM system.

In the following example, note that the category "Mechanical Pencils" occurs in more than one place in the hierarchy. This type of hierarchy is not ideal for PIM ICC setup;
use of this structure will result in lower data quality and higher costs related to maintaining data quality:

**Relationship Between ICC and Semantic Model**

There is a one-to-one relationship between the ICC structure and the data lens generated from the Semantic Model. Since a product hierarchy ICC structure ensures that any given item will have only one Semantic Model, the process of standardizing data is automated, accurate, and reproducible. If a PIM system were to contain more than one ICC for a given category, then multiple Semantic Models and possibly multiple and conflicting standardization rules could exist for a given item. This can be successfully avoided by upholding the one-to-one relationship between the ICC structure and the Semantic Models, a natural result of using a product hierarchy, as previously described.
Cleansing and Matching Item On-Boarding Process

The integration of the EDQP solution with the Fusion PIMDH system adds an important set of data quality capabilities every time product data is imported using the Import Workbench into PIM. The set of pre-built integration processes allow for various automated data quality checks and operations such as:

- Automated Categorization of data into the correct ICC.
- Automated assignments to alternate catalogs categories thru ICC and attribute mappings.
- Automated extraction, validation and standardization of category specific attributes.
- Automated multiple description generation and standardization from category specific attributes.
- Duplicate identification of incoming data.
- Duplicate prevention and cross-reference of new items to existing items in the production tables.
- Exception identification and routing.
- Real time and batch data quality checks from the Fusion PIMDH system and processes.
This chapter provides information you need to get started implementing EDQP to work with Fusion PIMDH, including prerequisites, preparing to implement, and the components in the product package.

Getting Started

To use all of the EDQP for Fusion PIMDH functions, you must ensure that you have completed the following actions before implementing, configuring, or using it:

1. Install the Fusion PIMDH server (including the latest patch sets)
2. Install and configure an EDQP 11g (11.1.1.6) Oracle DataLens Server. For instructions, see Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Installation Guide.
   The Fusion PIMDH product is included in the EDQP installation as a zip file.
3. Install and configure the EDQP Services for Excel 11g (11.1.1.6). For instructions, see Oracle Enterprise Data Quality for Product Data Services for Excel Reference Guide.

Preparing for Implementation

Locate and unzip the Fusion PIMDH release 11g (11.1.1.6) product:

1. Log in to your Oracle DataLens Server system as your administrator user.
2. Locate the Middleware home directory into which you installed EDQP. The default directories are the following:
   - On Linux and UNIX:
     /opt/Middleware/Oracle
   - On Windows:
     C:\Oracle\Middleware
   This directory is referenced as MW_HOME in Fusion Middleware documentation.
3. Unzip MW_HOME/edqp_template1/opdq-connectors-11-1-1-6.zip in the same directory to extract all files.
4. Create a directory named FUSION_PIM.
5. Unzip the opdq-connector-fusion_pim.zip file to the FUSION_PIM directory to extract all files.
EDQP for Fusion PIMDH Package Contents

The EDQP for Fusion PIMDH product package contains the following:

- opdq-connector-fusion_pim.zip - DSA and data lens files in the autodeploy directory structure:

```
   \---autodeploy
      \---dsa
          PRODUCT_HUB_DSA.xm
      \---lens
          opdq_flec_cons_pl.project
          Generic_Catch_ALL_DataLens.project
```

Implementing the EDQP for Fusion PIMDH Files

Implement the EDQP for Fusion PIMDH application DSA and data lens files:

1. On your Oracle DataLens Administration Server system, log in using the administrator user you established when installing the server in preparation to copy files.

   **Note:** This step does *not* refer to the Oracle DataLens Administration Server Web page.

2. Copy the $MW_HOME\edqp_template1\autodeploy directory that you extracted from the product package to the WebLogic dls_domain directory on your Oracle DataLens Administration Server. By default, these directories are:

   On Linux and UNIX: $MW_HOME/user_projects/domains/dls_domain/opdq/autodeploy

   On Windows: $MW_HOME\user_projects\domains\dls_domain\opdq\autodeploy

   **Note:** The preceding directories are the defaults when installing your Oracle DataLens Server; your installation directory may vary.

   This automatically deploys (autodeploys) the DSA and data lens files to your Oracle DataLens Server. The Oracle DataLens Administration Server polls the autodeploy directory every 10 minutes and attempts to autodeploy the DSA and data lens files that are placed in the respective folders under the autodeploy parent folder.

3. Log out of the Oracle DataLens Administration Server system.

4. Wait 10 minutes to ensure that the DSA and data lens files have been autodeployed.

5. Log into the Oracle DataLens Server Administration Web page.

   For more information about these Web pages, see *Oracle Enterprise Data Quality for Product Data* Oracle DataLens Server Administration Guide.

6. Verify that the DSAs and data lens have been autodeployed.
Establishing the EDQP Oracle DataLens Server Connection

After implementing Fusion PIMDH, use the Oracle Enterprise Manager to establish or modify the connection with the Oracle DataLens Server. For more information, see Oracle Fusion Applications Post-Installation Guide at:

http://docs.oracle.com/cd/E28271_01/fusionapps.1111/e22380/toc.htm

Enabling Fusion PIMDH DQ Processing

After you have installed the necessary EDQP components, you must return to PIM and perform the following additional configuration actions that are described in detail in the Item Data Quality Management white paper (see “Related Documents” on page 2-v):

■ Configure PIM to enable Data Quality (DQ) processing and interface with EDQP.
■ Configure the DQ ICC verification profile options.
■ Optional - Define the DQ Alternate Catalog to be used for data processing.
■ Define the Item Class Catalogs (ICC) that you want to use including valid values for attribute groups and attributes.
■ For each ICC, set up the DQ processing parameters by defining the attributes that will participate.
This chapter describes how to use the Fusion PIMDH DQ metadata extracts to create or update EDQP data lens to process your data.

**Metadata and Data Lens Synchronization Process**

Using ICC metadata extracted from Fusion PIMDH and Autobuild, you can create or update an EDQP data lens that will be used to standardize and classify your PIM data.

AutoBuild constructs a data lens by examining the available product metadata. Given sufficient information, AutoBuild can accomplish the following.

1. Construct a full Item Definition hierarchy, complete with required, scoring, and optional attributes
2. Construct rich term and phrase recognition rules
3. Provide an initial set of classification rules

AutoBuild offers a familiar, easy-to-use graphical wizard interface that guides you through the process from start to finish.

The Fusion PIMDH is used to enable DQ and configure certain parameters. The ICC metadata and sample data to be used for testing is exported. The metadata is imported into Excel and then used to build a semantic model (data lens) using the EDQP AutoBuild application. The resulting data lens is then refined and the *Product_Hub*_
DSA DSA is updated so that it uses your refined data lens to process your DQ metadata. The following diagram depicts this process:

---

**Exporting ICC Metadata from Fusion PIMDH**

The first step in either creating a data lens or synchronizing an existing data lens is to export the ICC metadata from Fusion PIMDH. This export shall include the Item Class as well as all the relevant sub classes. The creation of the following three exports is described in the *Item Data Quality Management* white paper (see "Related Documents" on page 2-v); each is a tab-delimited text file to be imported into an Excel workbook. The information contained in this PIMDH Excel export workbook is used to create or update a data lens. The following sections describe how the fields in the export workbooks are used in the data lens.

The following represents a portion of a Fusion PIMDH metadata worksheet extract:

---

**Item Class & Attribute Metadata Export**

The following table and illustrations describe how each of these fields is used in a data lens including explanations of how the exported metadata is used in a data lens:
<table>
<thead>
<tr>
<th>Metadata</th>
<th>Description</th>
<th>Excel Column</th>
<th>Data Lens Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Catalog Category Code</td>
<td>Unique identifier in Fusion PIM for the parent item class catalog category.</td>
<td>PARENTCATALOGCODE</td>
<td>The values in these columns are used to define the Item Definition structure within the data lens. The Item Definition structure is like the ICC structure in PIMDH.</td>
</tr>
<tr>
<td>Parent Catalog Category Name</td>
<td>The name of the parent Item Class Catalog.</td>
<td>PARENTCATALOGNAME</td>
<td></td>
</tr>
<tr>
<td>Catalog Category Code</td>
<td>New unique identifier in Fusion PIM for the ICC</td>
<td>CATALOGCODE</td>
<td>Illustrated in &quot;Item Definition Structure&quot; on page 3-4.</td>
</tr>
<tr>
<td>Catalog Category Name</td>
<td>The name of the ICC.</td>
<td>CATALOGNAME</td>
<td></td>
</tr>
<tr>
<td>Attribute Id</td>
<td>The Groupcode.Attributecode; the default separator is a period(.)</td>
<td>ATTR_ID</td>
<td>The value in this column is stored in the Attribute Alias field in the Item Definition. It is used to track attributes and their values through the synchronization process and should not be modified. Illustrated in &quot;Valid Value Attributes, Phrase Rules, and Term Rules&quot; on page 3-5.</td>
</tr>
<tr>
<td>Attribute Name</td>
<td>The name of the attribute.</td>
<td>ATTR_NAME</td>
<td>This value performs two functions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ As an attribute name in the Item Definition.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ As a phrase rule in the data lens.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Illustrated in &quot;Valid Value Attributes, Phrase Rules, and Term Rules&quot; on page 3-5.</td>
</tr>
<tr>
<td>Attribute Value:</td>
<td>Valid values are compared with distinct values from the item master; when there are no valid values or any records in the item master with values for an attribute, a row is created for the attribute containing a null value. Items associated with a unit of measure (UOM), the number is joined with the UOM into a single field.</td>
<td>ATTR_VALUE</td>
<td>This value is used to create a term rule in the data lens.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Illustrated in &quot;Valid Value Attributes, Phrase Rules, and Term Rules&quot; on page 3-5.</td>
</tr>
<tr>
<td>Attribute Base UOM Code:</td>
<td>For non-UOM based attributes, this value is null</td>
<td>ATTR_BASE_UOM_CODE</td>
<td>This value specifies the unit of measure for the attribute.</td>
</tr>
<tr>
<td></td>
<td>For UOM based attributes, this value is populated with the UOM code</td>
<td>ATTR_BASE_UOM_VALUE</td>
<td>This is the standardization value for the unit of measure in the column ATTR_BASE_UOM_CODE. This is the value that will be loaded into PIMDH. Illustrated in &quot;Units of Measure Values and Standardizations&quot; on page 3-5.</td>
</tr>
</tbody>
</table>
Exporting ICC Metadata from Fusion PIMDH

The following figures illustrates how the data lens is constructed based on the metadata column values described in the previous table:

**Item Definition Structure**

```plaintext
<table>
<thead>
<tr>
<th>Metadata</th>
<th>Description</th>
<th>Excel Column</th>
<th>Data Lens Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid Value Code</td>
<td>For non-Valid Value Based attributes, this value is null</td>
<td>VALID_VALUE_CODE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>For Valid Value Based attributes, this value is the Valid Value Code</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value Set Name</td>
<td>For non-Valid Value Based attributes, this value is null</td>
<td>VALUESET</td>
<td>If an attribute has a value set associated with it, the value in this column specifies the name of the value set. This value is used to create the value set phrase rule. The ATTR_VALUES associated with a VALUESET are used to create valid value term rules. For more information about these attribute values, see &quot;Using Value Sets when Creating a Data Lens&quot; on page 4-4.</td>
</tr>
<tr>
<td></td>
<td>For Valid Value Based attributes, this value is the Value Set Name associated with the attribute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attribute Match Indicator</td>
<td>Y or N: The appropriate metadata from the attribute match</td>
<td>ATTR_MATCH</td>
<td>The values in this column govern whether or not an attribute participates in the configuration of the Exact_Match match rules. Illustrated in &quot;Match Attributes&quot; on page 3-6</td>
</tr>
<tr>
<td>Language</td>
<td>The base language of the user running the export expressed as a Java Locale entry. For example, for US English the correct entry is 'en_US'</td>
<td>LANGUAGE</td>
<td>This describes the base language of the export.</td>
</tr>
</tbody>
</table>

**Note:** Some of your Fusion PIMDH DQ attribute values may be effectively recognized by the DLS_Import_Template rules; if this is the case, then the data lens generated by AutoBuild may also contain term and phrase rules directly from DLS_Import_Template. For more information about AutoBuild, see Oracle Enterprise Data Quality for Product Data AutoBuild Reference Guide.
Valid Value Attributes, Phrase Rules, and Term Rules

Units of Measure Values and Standardizations
The value from the `ATTR_BASE_UOM_CODE` is used to select the target in the UOM conversion for the attributes specified. For this value to be selected as the target, it must first exist as a term variant within the DLS_Import_Template Smart Glossary. The value in the `ATTR_BASE_UOM_VALUE` is used to set the standardization rules in Match Attributes. The following figures detail the relationship between the values seen in the metadata file and the corresponding features found in the data lens. Metadata export containing unit of measure values:

In the data lens created by AutoBuild, the `ATTR_BASE_UOM_CODE` is one of the term variants in the "[microfarad]" rule:
In the Unit Conversion tab, the microfarad (uF) is the target unit of measure for the "a_capacitance" phrase rule:

AutoBuild sets the standardization value in **Match_Attributes** for the term rule "[microfarad]" based on the metadata value from the **ATTR_BASE_UOM_VALUE** column that contained the microfarads:

**Match Attributes**

Required match attributes with "Y" in the **ATTR_MATCH** column are set as required in the standardization type **Match_Attributes** and match type **Exact_Match**:
Alternate Catalog Export

Using the Autobuild process, an Alternate Catalog can be added to a data lens by modifying an existing data lens. The Alternate Catalog Fusion PIMDH DQ spreadsheet export contains the following fields:

<table>
<thead>
<tr>
<th>Metadata</th>
<th>Description</th>
<th>Excel Column Header</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog Name</td>
<td>The name of the alternate catalog</td>
<td>CATALOG</td>
</tr>
<tr>
<td>Parent Category Code</td>
<td>The parent category code if not top-level category</td>
<td>PARENT_CATEGORY_ID</td>
</tr>
<tr>
<td>Parent Category Name</td>
<td>The parent category name if not top-level category</td>
<td>PARENTCategory_DESCRIPTION</td>
</tr>
<tr>
<td>Catalog Category Code</td>
<td>The category code.</td>
<td>CATEGORY_ID</td>
</tr>
<tr>
<td>Catalog Category Description</td>
<td>The name of the ICC</td>
<td>CATEGORY_DESCRIPTION</td>
</tr>
<tr>
<td>Language</td>
<td>The base language of the user running the export expressed as a Java Locale entry. For example, for US English the correct entry is 'en_US'</td>
<td>LANGUAGE</td>
</tr>
</tbody>
</table>

Given a Fusion PIMDH Catalog Name "Purchasing Catalog", the following classification type is added to the data lens:
Importing a Metadata File into Excel

Item Class Sample Item Export

The resulting export should contain 100 rows of item descriptions that hierarchically belong to the specified ICC parameter. This sample data extract is used as a data sample in the EDQP Knowledge Studio to test and refine the data lenses created using AutoBuild.

<table>
<thead>
<tr>
<th>Metadata</th>
<th>Description</th>
<th>Excel Column Header</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICC Name</td>
<td>The item name</td>
<td>ITEM_CLASS</td>
</tr>
<tr>
<td>ICC Data</td>
<td>The item data</td>
<td>ITEM</td>
</tr>
<tr>
<td>ICC Description</td>
<td>The item description</td>
<td>DESCRIPTION</td>
</tr>
</tbody>
</table>

Importing a Metadata File into Excel

Each Fusion PIMDH DQ text file exported must be imported into Excel for use in the metadata synchronization process as follows:

1. Open Excel.
2. Ensure that a blank workbook is open so that Services for Excel can populate it with the imported data otherwise an error occurs.
3. Click the Add-Ins tab to access the Services for Excel toolbar.
4. Click the DataLens Tools menu then select Load Meta-Data Files....
5. Locate the exported Fusion PIMDH DQ text file, select it, and then click Open.

This utility converts the text file into an Excel spreadsheet with the correct column headers required by Autobuild. It ensures that the contents of the file are not corrupted by the standard Excel import process, such as automatically converting some fields to dates or deleting leading zeros.
6. Save the Excel workbook.

Creating a Data Lens

A new data lens can be created using AutoBuild from an Item Class & Attribute Metadata metadata export that has been imported into Excel using the following steps:

1. Ensure that you have imported your metadata into an Excel workbook using the steps in “Exporting ICC Metadata from Fusion PIMDH” on page 3-2.
2. Open the Excel workbook that contains your imported metadata.
3. Click the Add-Ins tab to access the Services for Excel toolbar.
4. Click the AutoBuild button.
Step 1 of the AutoBuild wizard is displayed.

5. Select **Generate a new DataLens**.

Each step of the wizard is populated with the necessary options selected. For more information, see *Oracle Enterprise Data Quality for Product Data AutoBuild Reference Guide*.

6. Ensure that the options are selected or entered as follows then click **Next**:
   - Category Source section:
     - DLS_Import_Template from the Smart Glossary or Existing DataLens list
     - 1,2,3,4 is entered in the List of category columns field
     - Multi Column List of Category Code/Title Pairs in the Category column layouts subsection

Step 2 of the wizard is displayed.

7. Ensure the options are selected or entered as follows then click **Next**:
   - Required Attributes section:
     - Add Required Attribute
     - 4 is entered in the Item Name Column field
   - Scoring Attributes section:
     - Add Scoring Attributes
     - Attribute names are in the same row as the category
     - Row contains Attribute Name/Value/UOM
     - 6,7,8 is entered in the Scoring Attributes field
     - 6,7,8 is entered in the Single term columns field

Step 3 of the wizard is displayed.

8. Ensure that the **Use DataLens Knowledge Studio to collect samples** option is selected then click **Next**:

Step 4 of the wizard is displayed including the Item Definition structure that will be created in the data lens.

9. Ensure the options are selected as follows:
   - DataLens Options section:
     - **Generate / Update Data Lens**
     - The data lens name displayed in the **New/Updated Data Lens** list corresponds to the top level ICC in your metadata Excel workbook; your top level Item Definition will also have this name.

10. Click **Output Details** to display the corresponding options.

11. Ensure the Output Details options are selected as follows:
   - Generate Report check box is selected
   - **Use Alias for Ids** check box is selected

12. Click **Finish**.
The new data lens is created from the category and attribute information provided in your structured metadata. When the AutoBuild process completes, you are returned to your worksheet.

13. Start Knowledge Studio and open the data lens you just created.
For more information, see Oracle Enterprise Data Quality for Product Data Getting Started Guide and Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide.

14. Examine your new data lens and refine the data recognition rules to add variants that may not have been automatically generated. See “Refining Your New Data Lens” on page 4-1.

15. Save your data lens, and then check it in and deploy it to your Oracle DataLens Server.

**Updating an Existing Data Lens**

Using AutoBuild and the following process, an existing data lens can be updated from an Item Class and Attribute Metadata or an Alternate Catalog metadata export that has been imported into Excel using the following steps:

---

**Note:** The data lens being updated must be checked out and locked for editing by the user running the update process or the data lens cannot be updated. If the data lens is checked out read-only, it cannot be updated.

---

1. Ensure that you have imported your metadata into an Excel workbook using the steps in "Exporting ICC Metadata from Fusion PIMDH" on page 3-2.
2. Open the Excel workbook that contains your imported metadata.
3. Click the Add-Ins tab to access the Services for Excel toolbar.
4. Click the AutoBuild button.
   Step 1 of the AutoBuild wizard is displayed.
5. Select Full update of existing DataLens.
   Since each step of the wizard is pre-populated with PIM-specific selections, it will generally be unnecessary to make any changes to the selections. For more information, see Oracle Enterprise Data Quality for Product Data AutoBuild Reference Guide.
6. Ensure that the options are selected or entered as follows then click Next:
   - Category Source section:
     - Select the data lens that you want to update fully from the Smart Glossary or Existing DataLens list
     - 1,2,3,4 is entered in the List of category columns field
     - Multi Column List of Category Code/Title Pairs in the Category column layouts subsection
   Step 2 of the wizard is displayed.
7. Ensure the options are selected or entered as follows then click Next:
   - Required Attributes section:
Adding an Alternate Catalog to a Data Lens

If you use an Alternate Catalog, it must be added to all the data lenses that are going to participate in the Fusion PIMDH DQ process.

Use the following steps to add an Alternate Catalog to an existing data lens:

1. Open the Excel workbook that contains your imported Alternate Catalog metadata.
2. Click the Add-Ins tab to access the Services for Excel toolbar.
3. Click the AutoBuild button.

Add Required Attribute
– 4 is entered in the Item Name Column field

Scoring Attributes section:
– Add Scoring Attributes
– Attribute names are in the same row as the category
– Row contains Attribute Name/Value/UOM
– 6,7,8 is entered in the Scoring Attributes field
– 6,7,8 is entered in the Single term columns field

Step 3 of the wizard is displayed.

8. Ensure that the Use DataLens Knowledge Studio to collect samples option is selected then click Next:

Step 4 of the wizard is displayed including the Item Definition structure that will be created in the data lens.

9. Ensure the options are selected as follows:

DataLens Options section:
– Generate / Update Data Lens
– The data lens name displayed in the New/Updated Data Lens list should be the top-level Item Definition displayed in the Item Definition structure.

Processing begins then the Step 4 of the wizard is displayed.

10. Click Finish.

The new data lens is created from the category and attribute information provided in your structured metadata. When the AutoBuild process completes, you are returned to your worksheet.

11. Start Knowledge Studio and open the data lens you just created.

For more information, see Oracle Enterprise Data Quality for Product Data Getting Started Guide and Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide.

12. Examine your new data lens and refine the data recognition rules to add variants that may not have been automatically generated. See "Refining Your New Data Lens" on page 4-1.

13. Save your data lens, and then check it in and deploy it to your Oracle DataLens Server.

Adding an Alternate Catalog to a Data Lens

If you use an Alternate Catalog, it must be added to all the data lenses that are going to participate in the Fusion PIMDH DQ process.

Use the following steps to add an Alternate Catalog to an existing data lens:

1. Open the Excel workbook that contains your imported Alternate Catalog metadata.
2. Click the Add-Ins tab to access the Services for Excel toolbar.
3. Click the AutoBuild button.
Changing the Data Lens in the DQ Processing DSA

Step 1 of the AutoBuild wizard is displayed.

Each step of the wizard is populated with the necessary options selected. For more information, see Oracle Enterprise Data Quality for Product Data AutoBuild Reference Guide.

4. Select **Add alternate catalog to Existing DataLens**.

5. Select the existing data lens.

6. Click **Next**.

   The data is automatically processed and Step 4 of the wizard is displayed including the Item Definition structure that will be created in the data lens.

7. Click **Output Details** to display the corresponding options.

8. Ensure that the **Use Alias for Id's** check box is selected in the Output Details section then click **Finish**.

   The new data lens is created from the category and attribute information provided in your structured metadata. When the AutoBuild process completes, you are returned to your worksheet.

9. Start Knowledge Studio and open the data lens you just created.

   For more information, see Oracle Enterprise Data Quality for Product Data Getting Started Guide and Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide.

   When using an alternate catalog for DQ, you must map the Item Definitions (Fusion PIMDH Item Classes) to the appropriate alternate catalog categories in your data lens using the Knowledge Studio. For more information, see Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide.

10. Select the **Classify** tab.

11. From the **Classification Types** list, select the Alternate Catalog you just created using AutoBuild.

   The data lens updates how the sample data is classified using the new Alternate Catalog and redispays the data.

12. Review the categories to determine the categories that should be mapped with the Fusion PIMDH ICCs.

13. Map an ICC to a category by dragging the Item Definition icon to the correct category of the catalog in the **Classification Type** pane.

14. Repeat the mapping process for all Item Definitions that should be mapped to a category.

15. Save your data lens, and then check it in with deployment to your Oracle DataLens Server.

### Changing the Data Lens in the DQ Processing DSA

The Fusion PIMDH is automatically configured to use the DSA and data lenses delivered with the product. The DSA template must be changed to use a data lens developed for your environment using the metadata synchronization process.

**Note:** Only an EDQP Administrator should change the DSA to avoid configuring it improperly causing the inability to process data.
1. Start the EDQP client software.

2. Start Application Studio. For more information, see Oracle Enterprise Data Quality for Product Data Application Studio Reference Guide.

3. Select Product_Hub_DSA from the list.
   If this DSA does not appear in the Open Data Service Application dialog list, that means it is not checked out. Follow these steps to check it out:
   a. Click Cancel to close the Open Data Service Application dialog.
   b. Click the Check-Out button.
   c. Select Product_Hub_DSA from the list.
   d. Select the Lock server file for editing option so that the DSA can be edited.
   e. Click OK.

4. Double-click the Extract_Attributes step to open the Transform Map.

5. Double-click on the Perform_DQ_Checks Item Definition transform to open it.

6. Click the Select Data Lenses button.

7. From the Deployed Data Lenses list, double-click the data lens you want to use to move it to the Selected Data Lenses list.

8. From the Selected Data Lenses list, select your data lens then click the up arrow until it is the first data lens in the list.

   **Note:** Your data lens must be above the Generic_Catch_All_DataLens data lens in the Selected Data Lenses list or the process will not operate correctly.

   **Note:** If this is the first time you have modified the Product_Hub_DSA DSA, you must remove the DQ_Elec_Comp_AG_DL sample data lens shipped with the product to ensure that the real time Fusion PIMDH DQ operates correctly.

9. Click OK to close the dialog then click OK when the informational message is displayed.

10. Click OK to close the Item Definition dialog.

11. Close and save the Transform Map.

12. Save the DSA, and then check it in with deployment to your Oracle DataLens Server.

### Refreshing Metadata and Semantic Keys

Numerous refresh cycles can occur on a PIM system that would require you to rerun the "Exporting ICC Metadata from Fusion PIMDH" on page 3-2 and "Updating an Existing Data Lens" on page 3-10 processes. The following are a few that should be considered:

**Refreshing metadata after a change to an ICC structure**

- Adding an ICC
- Deleting an ICC
- Renaming an ICC
- Adding a new attribute to an existing ICC
- Deleting an attribute from an existing ICC
- Renaming an attribute from an existing ICC
- Adding a new valid value to an existing attribute
- Deleting a valid value from an existing attribute

**Changing Attributes that participate in a Match**
- Adding more attributes to a match definition
- Removing attributes from a match definition
- Refreshing the Semantic Keys for items in PIM Production
This chapter describes how to maintain your EDQP DSAs and data lens using the real time synchronization with Fusion PIMDH. These best practices have been developed so that you can easily ensure that your changing data is processed correctly.

Refining Your New Data Lens

The data lens generated by Autobuild contains the structure you need for the PIM metadata synchronization. The Item Definitions, attributes, phrases, and terms have been created from the PIM export metadata; term rules and standardization rules can be modified in order to adequately handle the variability seen in production data.

In order to keep your data lens synchronized with the PIM metadata, there are editing restrictions on certain data lens elements: All additions, deletions, and renames should be done only in synchronization with Fusion PIMDH. Unless your metadata has changed, and you are performing deletions based on changes reported in the Semantic Knowledge Structure report, these guidelines should be followed:

- Neither add nor delete Item Definitions.
- Neither add nor delete attributes from any Item Definitions.
- Do not modify the Item Definition Attribute Alias. If the alias is modified, the standardized attribute value cannot be loaded into PIMDH.
- Value set guidelines:
  - Do not add any values to the value set phrase rules.
  - It is permissible to add term variants to the valid value term rules.

The following is an overview of recommendations for refining your data lens in the EDQP Knowledge Studio:

1. Ensure that the required attributes in the Item Definition are being recognized.

   If the ICC name is not identical to the item name found in product data, use the following steps in Knowledge Studio:

   a. Add term and phrase rules to your data lens.
b. Associate the new phrase rule with the appropriate required attribute in your Item Definition.

2. Review and resolve ambiguities.

If there are any recognition or Item Definition ambiguities in your sample data, review and resolve them.

3. Increase recognition coverage.

a. Using your data lens sample data files, apply the standard data lens development methodology for completing recognition as appropriate.

b. You may need to increase the coverage of your data lens to account for variants of attribute values found in your sample data. As you review your sample data files, examine the unrecognized text. If you encounter any tokens which should be associated with terms and attributes in your data lens, add them. In the following example, the selected line, "AL" is recognized as being a member of a valid value within a value set. In the following line of data the token "Aluminum" is unrecognized.
Since "Aluminum" is a variant of the valid value "AL", the token "Aluminum" should be added as a term variant to the term rule [al_vv]:

4. Review Units of Measure.

As previously described, if a unit of measure is specified in the column ATTR_BASE_UOM_CODE that is not present in the data lens, then it must be added to the appropriate term rule because AutoBuild does not add it. Then, you should examine the relevant Item Definition attribute and associate the modified UOM phrase rule with it.

5. Review term and phrase standardizations.

a. Examine the standardized attribute values in the tabs Standardize Items | Test Attributes.

Term Standardization - Many of the term standardization rules in the Match_Attributes Standardization Type may have been provided by the Smart Glossary import used in the AutoBuild process. Review these standardization rules and either keep or modify them.

Phrase concatenation - Some of your multi-term attribute values may require concatenation.

**Note:** If you selected the Create Replace all Rules option in AutoBuild, the preceding standardization occurs automatically.


Review the match type and match rules associated with Match_Attributes. The Exact_Match rules have been created based on data from the metadata export.

7. Activation of Item Definitions.

By default, the Item Definitions are inactive upon finishing the Autobuild process. Activate your Item Definitions before running a DSA that processes data through your data lens.

8. Refine Item Definitions.

Use search logic and value logic as necessary. The use of search logic is necessary to constrain the attribute values in the following example. The following line of data contains two temperature values:

Capacitor, AL, 22uF, 10V, 50 deg C, 100 deg C, 1%, RD, .10S, 10mm Dx 16mm L

Use of value logic in the Item Definition is necessary to correctly associate the minimum and the maximum temperature values with the correct attributes.
9. If you have added an Alternate Catalog, classify your data.

Using standard classification methodology, create classification rules by classifying from Item Definitions, Rules or Data.

10. After making your refinements, check in your data lens and deploy it to Production.

Using Value Sets when Creating a Data Lens

Certain attributes may have been set up in PIM to have value sets and valid values. The attribute values are restricted such that only the values specified are valid. When you create a Semantic Model, this value set and valid information is exported from the Fusion PIMDH and it appears in your metadata worksheet. The data lens created from the metadata will reflect the valid value and value set information from the metadata worksheet. If there is a value set for an attribute, the VALUESET column will be populated with the name of the value set, and the valid values appear in the attr_value column as in the following example:
There are two types of value sets: Standard Value Sets and Yes/No Value Sets. The following describe how EDQP handles them in a data lens:

**Standard Value Sets**

In a standard value set, the valid values correspond to the standardized forms of attribute values that are likely to be found in product data. In this example, 'Dielectric' is the attribute identified in the 'value set' column and 'AB', 'AL', 'CE', 'MP', and 'TA' in the 'attr_value' column. The value is considered to be an abbreviation unless it exceeds four characters, in which case, the fullform in the EDQP Knowledge Studio is set to this value. For example using the metadata in “Value Set Example Spreadsheet” on page 4-5, 'AL' would set the term as an abbreviation while 'Aluminum' would set it to the full form. The phrase structure for the abbreviation is:

**Yes/No Value Sets**

The Yes/No value set is a special form of the value set; this value set has 'yes' and 'no' (and alternate forms, such as 'y' and 'n') as the valid values. For example, using the metadata in “Value Set Example Spreadsheet” on page 4-5, the value set is a yes/no value set, and the valid values are 'yes' and 'no'. The data lens is created such that there are phrase and term rules to recognized forms of "shielded" and "not-shielded". Additionally, the standardization rules for these 'y' and 'n' attributes will be set to "yes" and "no". The phrase structure for the yes, or shielded, term is:

All variations of both yes and no are created in the data lens. For example, the variations of 'not shielded' would be:

- 'not shielded'
- 'not-shielded'
- 'not- shielded'
- 'not - shielded'
- 'notshielded'
- 'notshielded'
Using Value Sets when Creating a Data Lens

In the data lens, all value set phrase rule names are created with a 'vs' appended to the attribute and 'vv' appended to the term rule name so that they are easily identified in phrase structures.

Refining Value Sets in a Data Lens

Refining value set phrase structures in the Knowledge Studio is the same as described in the previous section. Since it is possible for some text nodes and term rules to be recognized by multiple rules, it is necessary to review those attribute in your data lens that are based on value sets. For example, the 'Aluminum' attribute could be a production of the 'a_dielectric_vs' phrase rule. When appropriate, it is important to refine your data lens value sets by adding the full forms for the valid value term rules which are abbreviations. For example, you will need to add the term variant 'aluminum' to the term rule 'al_vv'. Additionally, set the full form to be 'aluminum'. Using the 'Aluminum' example as in the following phrase rule:

You would edit the 'al_vv' term and add the 'Aluminum' full form (variants can be generated as well):

Then review the phrase rule to ensure that it was added properly:

Finally, review the productions of the 'a_dielectric_vs' phrase rule:
Updating Your Fusion PIMDH DQ Metadata Structure

Using Autobuild, you can synchronize your Fusion PIMDH metadata with your data lenses. It is possible that your metadata may change over time. You may have new categories, attributes, or attribute values or changes in attribute names. Attributes may be deleted from categories. Using Autobuild you can synchronize your Item Definitions with your ICCs.

After creating a new metadata import spreadsheet reflecting the changes in metadata, you run Autobuild and update an existing data lens thus revising with the changes. The following actions occur:

1. Autobuild reports on four types of changes in the metadata and this information is reported in the Semantic Knowledge Structure sheet in your metadata import workbook:
   - New metadata.
   - Renamed metadata. Entity existed in prior version of lens, and has received a new name in the current metadata
   - Existing metadata. No change compared to previous version.
   - Not present in metadata. Entity existed in previous version, but not in current version and thus should be deleted from data lens.

2. Autobuild makes the specified changes to category and attribute names in your data lens.

3. Autobuild adds the specified new category, attribute, or value information to your data lens.

4. You must manually delete any items not present in current metadata. Autobuild does not perform deletions to your data lens. Refer to the Autobuild Semantic Knowledge Structure report for guidance.

Additions

In the following example, the shaded cells represent new metadata:

![Image of a spreadsheet with shaded cells]

This additional information is represented in the Semantic Knowledge Structure report as follows. The attribute "Capacitor_package_size" has a new attribute value.
This report also reports new term rules as follows:

<table>
<thead>
<tr>
<th>New</th>
<th>Old</th>
</tr>
</thead>
<tbody>
<tr>
<td>[a_color_coding]</td>
<td>([blue])</td>
</tr>
<tr>
<td></td>
<td>([red])</td>
</tr>
<tr>
<td></td>
<td>([green])</td>
</tr>
<tr>
<td></td>
<td>([yellow])</td>
</tr>
</tbody>
</table>

Renames
The attribute 'Mount' was renamed to 'Mount_type' as in the following example:

- Renamed:
  - Mount_type
- Old:
  - [a_rd_vs]
  - [a_xc_vs]

If your PIM data contains renamed item definitions or attribute, be sure to confirm that these renames are reflected in the data lens. Manual renaming may be required.

Deletions
To determine if a deletion should be made in the data lens, the user should review the AutoBuild report and examine all items flagged as 'DELETE'. If a category, attribute, or
attribute value is not present in the current metadata export file, but was in the previous version of the metadata export, it is identified as a 'DELETE' in the AutoBuild report.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>[a_mtw_vu]</td>
<td>((a_mtw_vu))</td>
</tr>
<tr>
<td>[i_capacitor]</td>
<td>{{capacitor}}</td>
</tr>
</tbody>
</table>

In the preceding figure, there are two deletions. The first rule tagged for deletion represents a value set value that was absent from the current export of the Fusion PIMDH DQ metadata. Both the phrase rule and term rule must be manually deleted from the data lens as in the following:

The second rule tagged for deletion is '[i_capacitor]' and should remain in the data lens because this rule was manually added to the data lens in order to recognize the item in product data. This rule is used as the required item name attribute in the Item Definition, and thus should remain in the data lens.