

**Oracle® Enterprise Data Quality for Product Data**

Fusion PIM Integration Implementation and User's Guide

Release 5.6.2

**E27167-01**

January 2012

Oracle Enterprise Data Quality for Product Data Fusion PIM Integration Implementation and User's Guide, Release 5.6.2

E27167-01

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Primary Author: Lorna Vallad

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

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

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

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

UNIX is a registered trademark of The Open Group.

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

# Contents

<b>Preface</b> .....	v
<b>1 Introduction</b>	
<b>Integration with Oracle Fusion Product Data Hub</b> .....	1-2
Fusion PIMDH Configuration.....	1-2
ICC Structure as Product Hierarchy .....	1-2
Relationship Between ICC and Semantic Model.....	1-4
Cleansing and Matching Item On-Boarding Process.....	1-5
<b>2 Implementing and Configuring</b>	
<b>Getting Started</b> .....	2-1
<b>Preparing for Implementation</b> .....	2-1
EDQP for Fusion PIMDH Package Contents .....	2-2
<b>Implementing the EDQP for Fusion PIMDH Files</b> .....	2-2
<b>Establishing the EDQP Oracle DataLens Server Connection</b> .....	2-3
<b>Enabling Fusion PIMDH DQ Processing</b> .....	2-3
<b>3 Using the Fusion PIMDH DQ Synchronization Process</b>	
<b>Metadata and Data Lens Synchronization Process</b> .....	3-1
<b>Exporting ICC Metadata from Fusion PIMDH</b> .....	3-2
Item Class & Attribute Metadata Export .....	3-2
Alternate Catalog Export .....	3-7
Item Class Sample Item Export.....	3-8
<b>Importing a Metadata File into Excel</b> .....	3-8
<b>Creating a Data Lens</b> .....	3-9
<b>Updating an Existing Data Lens</b> .....	3-10
<b>Adding an Alternate Catalog to a Data Lens</b> .....	3-12
<b>Changing the Data Lens in the DQ Processing DSA</b> .....	3-13
<b>Refreshing Metadata and Semantic Keys</b> .....	3-14
<b>4 Continued DQ Improvement</b>	
<b>Refining Your New Data Lens</b> .....	4-1
<b>Using Value Sets when Creating a Data Lens</b> .....	4-5
Refining Value Sets in a Data Lens.....	4-6
<b>Updating Your Fusion PIMDH DQ Metadata Structure</b> .....	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.

You must have Enterprise DQ for Product client software installed on your computer.

## 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 PIMDH 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 Enterprise DQ for Productdocumentation 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 COM Interface Guide* provides information about installing and using the Oracle DataLens Server COM APIs.
- The *Oracle Enterprise Data Quality for Product Data Java Interface Guide* provides information about installing and using the Oracle DataLens Server Java APIs.
- The *Oracle Enterprise Data Quality for Product Data Application Studio Reference Guide* provides information about creating and maintaining Data Service Applications (DSAs).
- 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 Governance Studio Reference Guide* provides information about creating and maintaining Data Service Applications (DSAs).
- The *Oracle Enterprise Data Quality for Product Data Glossary* provides definitions to commonly used Enterprise DQ for Product technology terms.
- 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.
- The *Oracle Enterprise Data Quality for Product Data Task Manager Reference Guide* provides information about managing tasks created with the Task Manager or Governance Studio applications.

See the latest version of this and all documents listed at the Oracle Enterprise Data Quality for Product Data Documentation Web site at:

[http://download.oracle.com/docs/cd/E20593\\_01/index.htm](http://download.oracle.com/docs/cd/E20593_01/index.htm)

## Conventions

The following text conventions are used in this document:

<b>Convention</b>	<b>Meaning</b>
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

---

<b>Convention</b>	<b>Meaning</b>
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

---

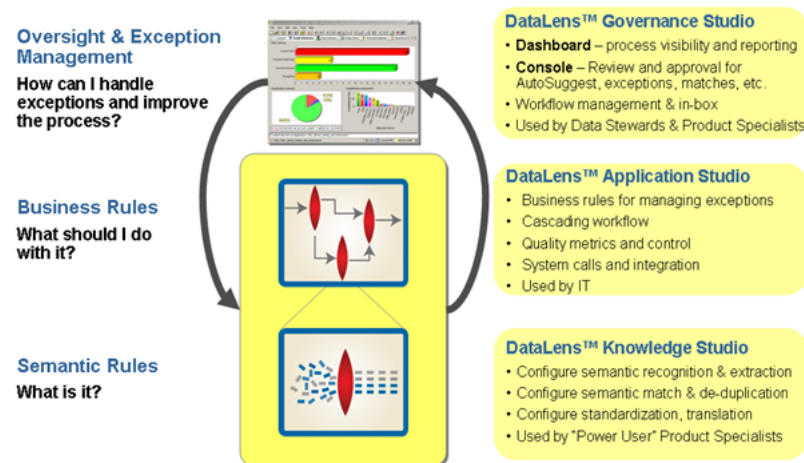




## Introduction

Oracle DataLens Server is built on industry-leading DataLens™ Technology to standardize, match, enrich, and correct product data from different sources and systems. The core DataLens Technology uses patented semantic technology designed from the ground up to tackle the extreme variability typical of product data.

Oracle Enterprise Data Quality for Product Data, formerly Oracle Product Data Quality, uses three core DataLens Technology modules: Governance Studio, Knowledge Studio, and Application Studio. The following figure illustrates the process flow of these modules.



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 arriving in 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 out of the box 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.

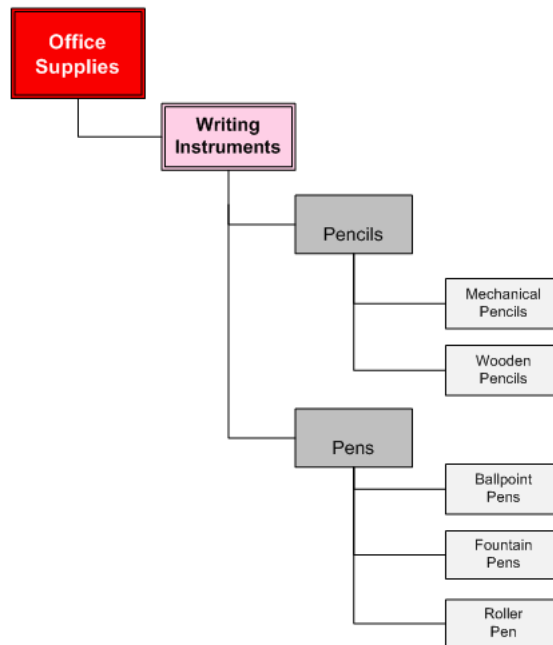
#### ICC Structure as Product Hierarchy

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

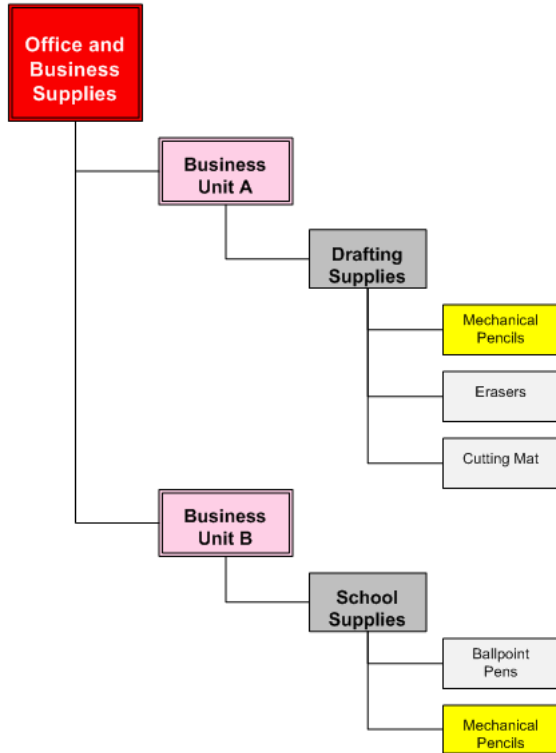
A product hierarchy contains multiple levels and each level is identified by a set of defining attributes. The levels must be non-overlapping and unique. Each item in PIM should belong to one and only one ICC. The category-specific descriptive attributes must be defined at the leaf-node level while the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

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



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.

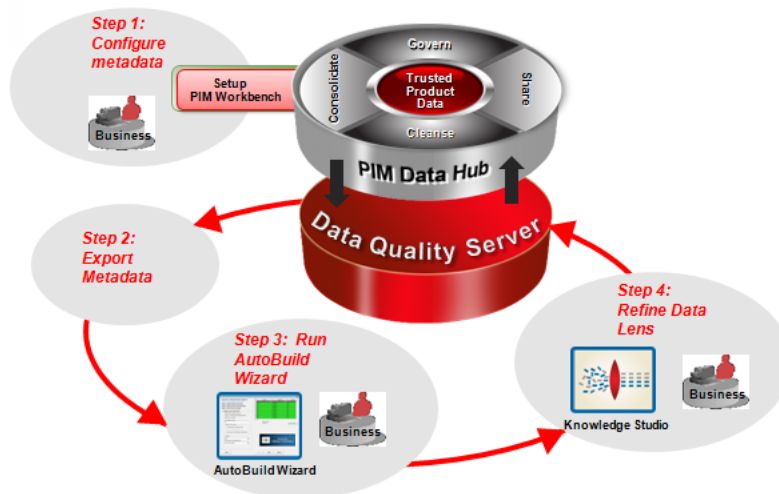
In the following example, note that the category "Mechanical Pencils" occurs in more than one place in the hierarchy. This type of hierarchy is not ideal for PIM ICC setup; use of this structure will result in lower data quality and higher costs related to maintaining data quality:



### 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.

### Metadata Synchronization Process

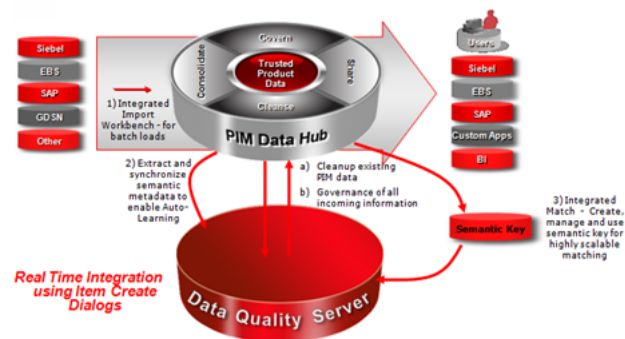


## Cleansing and Matching Item On-Boarding Process

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.

### EDQP + Product Hub = Integrated Product Functionality





---

---

## Implementing and Configuring

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 5.6.2 Oracle DataLens Server. For instructions, see *Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Installation Guide*.
3. Install the EDQP 5.6.2 client software. For installation instructions, see "[Installing the Client Software](#)" on page A-1.
4. Install and configure the EDQP Services for Excel 5.6.2. For instructions, see *Oracle Enterprise Data Quality for Product Data Services for Excel Reference Guide*.

### Preparing for Implementation

You must obtain the current EDQP for Fusion PIMDH product, which is delivered as a patch, from the Oracle Support Web site.

Locate and download the EDQP for Fusion PIMDH version 5.6.2 patch:

1. Browse to the My Oracle Support Web site:  
<https://support.oracle.com>
2. Log in or register.
3. Click the **Patches & Updates** tab.
4. Click **Product or Family (Advanced Search)**.
5. Search for Oracle Enterprise Data Quality for Product Data by beginning to enter the name in the **Product** field and then select it when it is displayed in the list.
6. Select the **Release** list, click the arrow adjacent to the Oracle Enterprise Data Quality for Product Data folder.
7. Select the **EDQP 5.6** release and then click **Close**.
8. Click **Search**.

9. Select the EDQP for Fusion PIMDH version 5.6.2 patch from the list of patches, and then click **Download**.
10. Click the patch file name, and then save it to a temporary directory.
11. Change directories to the temporary directory.
12. Unzip the patch zip file into the temporary directory.
13. Unzip the `autodeploy.zip` file into the temporary directory.

## EDQP for Fusion PIMDH Package Contents

The EDQP for Fusion PIMDH product package contains the following:

`dataLens_connector_fusion_5_6_x.zip` - DSA and data lens files in the following directory structure:

```

├── autodeploy.zip
├── dataLens_connector_fusion_5_6_x.zip
├── myFile.txt
├── readme.txt
└── autodeploy
    ├── dsa
    │   └── PRODUCT_HUB_DSA.xml
    └── lens
        ├── DQ_Elec_Comp_AG_DL.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 `autodeploy` directory that you extracted, from the patch set to a temporary directory, to the `local` directory of your Oracle DataLens Administration Server. By default, these directories are:

On Linux and UNIX: `/opt/Oracle/Middleware/opdq/local`

On Windows: `C:\Oracle\Middleware\opdq\local`

---



---

**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 `local` 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/E15586\\_01/fusionapps.1111/e22380/toc.htm](http://docs.oracle.com/cd/E15586_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.



## Using the Fusion PIMDH DQ Synchronization Process

This chapter describes how to use the Fusion PIMDH DQ metadata extracts to create or update EDQP data lens to process your data.

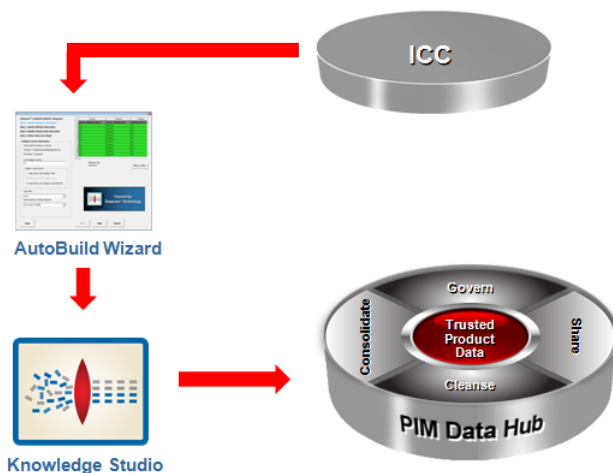
### Metadata and Data Lens Synchronization Process

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.

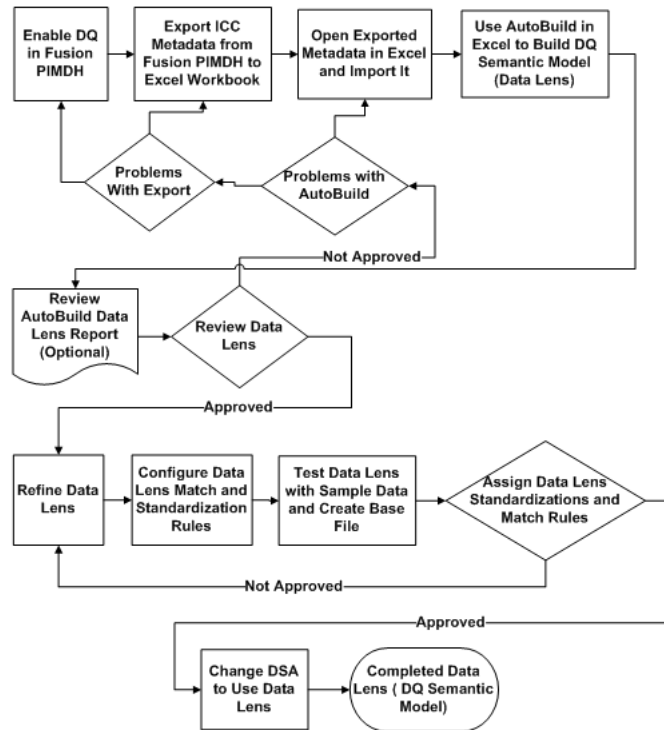
1. Construct a full Item Definition hierarchy, complete with required, scoring, and optional attributes
2. Construct rich term and phrase recognition rules
3. Provide an initial set of classification rules

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. Then the ICC metadata and sample data is exported. 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:

	A	B	C	D	E	F	G	H
1	PARENTCATALOGCODE	PARENTCATALOGNAME	CATALOGCODE	CATALOGNAME	ATTR_ID	ATTR_NAME	ATTR_VALUE	ATTR
2	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitance	Capacitance		1 uF
3	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	6.3mmdx11mmHg	
4	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	c case	
5	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	11.5mmdx14.5mml	
6	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	10mmdx16mml	
7	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	100mmdx23mml	
8	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	105mmdx28.5mml	
9	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	a case	
10	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	v case	
11	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ	Capacitors_AG.Capacitor_case_size	Capacitor_case_size	d case	

## 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:

Metadata	Description	Excel Column	Data Lens Function
Parent Catalog Category Code	Unique identifier in Fusion PIM for the parent item class catalog category.	PARENTCATALOGCODE	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.
Parent Catalog Category Name	The name of the parent Item Class Catalog.	PARENTCATALOGNAME	
Catalog Category Code	New unique identifier in Fusion PIM for the ICC	CATALOGCODE	Illustrated in " <a href="#">Item Definition Structure</a> " on page 3-4.
Catalog Category Name	The name of the ICC.	CATALOGNAME	
Attribute Id	The Groupcode.Attributecode; the default separator is a period(.).	ATTR_ID	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 " <a href="#">Valid Value Attributes, Phrase Rules, and Term Rules</a> " on page 3-5.
Attribute Name	The name of the attribute.	ATTR_NAME	This value performs two functions: <ul style="list-style-type: none"> <li>As an attribute name in the Item Definition.</li> <li>As a phrase rule in the data lens.</li> </ul> Illustrated in " <a href="#">Valid Value Attributes, Phrase Rules, and Term Rules</a> " on page 3-5.
Attribute Value:	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.	ATTR_VALUE	This value is used to create a term rule in the data lens.  Illustrated in " <a href="#">Valid Value Attributes, Phrase Rules, and Term Rules</a> " on page 3-5.
	Attribute Base UOM Code: <ul style="list-style-type: none"> <li>For non-UOM based attributes, this value is null</li> <li>For UOM based attributes, this value is populated with the UOM code</li> </ul>	ATTR_BASE_UOM_CODE  ATTR_BASE_UOM_VALUE	This value specifies the unit of measure for the attribute.  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 " <a href="#">Units of Measure Values and Standardizations</a> " on page 3-5.

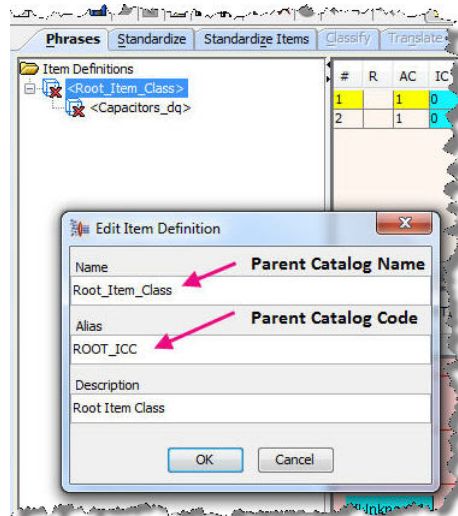
Metadata	Description	Excel Column	Data Lens Function
	Valid Value Code: <ul style="list-style-type: none"> <li>For non-Valid Value Based attributes, this value is null</li> <li>For Valid Value Based attributes, this value is the Valid Value Code</li> </ul>	VALID_VALUE_CODE	
	Value Set Name: <ul style="list-style-type: none"> <li>For non-Valid Value Based attributes, this value is null</li> <li>For Valid Value Based attributes, this value is the Value Set Name associated with the attribute</li> </ul>	VALUESET  For more information about these attribute values, see <a href="#">"Using Value Sets when Creating a Data Lens"</a> on page 4-5.	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 <a href="#">"Using Value Sets when Creating a Data Lens"</a> on page 4-5.
	Attribute Match Indicator: <ul style="list-style-type: none"> <li>Y or N: The appropriate metadata from the attribute match</li> </ul>	ATTR_MATCH	The values in this column govern whether or not an attribute participates in the configuration of the <b>Exact Match</b> match rules.  Illustrated in <a href="#">"Match Attributes"</a> on page 3-7
Language	<ul style="list-style-type: none"> <li>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'</li> </ul>	LANGUAGE	This describes the base language of the export.

**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.

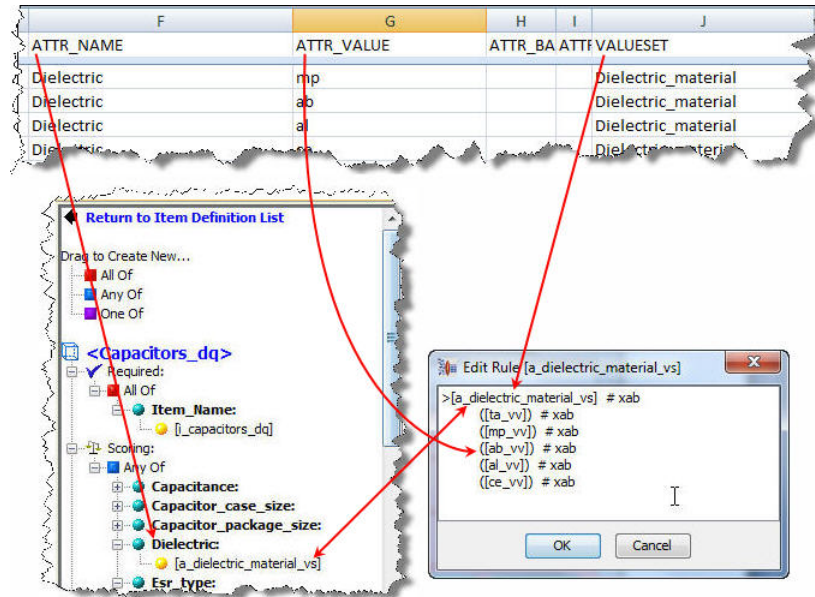
The following figure illustrates how the data lens is constructed based on the metadata column values described in the previous table:

**Item Definition Structure**

	A	B	C	D
1	PARENTCATALOGCODE	PARENTCATALOGNAME	CATALOGCODE	CATALOGNAME
2	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ
3	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ
4	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ
5	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ
6	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ
16	ROOT_ICC	Root Item Class	Capacitors_DQ	Capacitors_DQ



**Valid Value Attributes, Phrase Rules, and Term Rules**

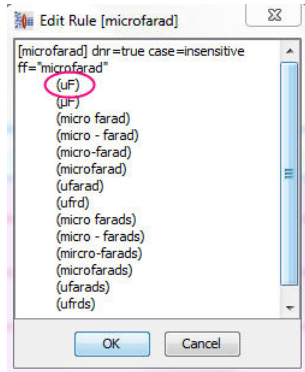


**Units of Measure Values and Standardizations**

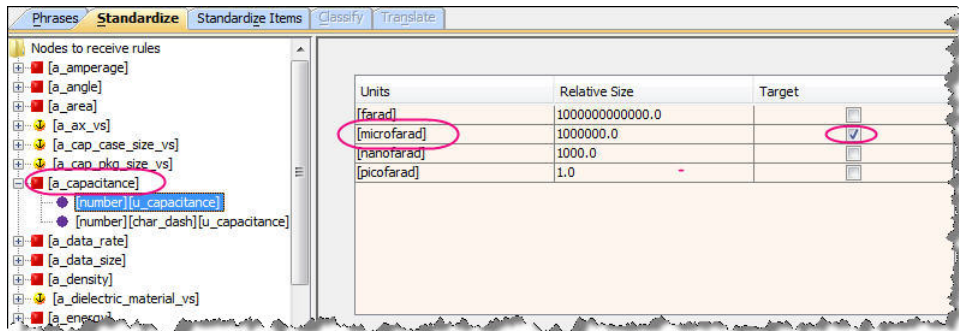
The value from the ATTR\_BASE\_UOM\_CODE is used to select the target in the UOM conversion for the attributes specified. For this value to be selected as the target, it must first exist as a term variant within the DLS\_Import\_Template Smart Glossary. The value in the ATTR\_BASE\_UOM\_VALUE is used 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:

F	G	H	I
ATTR_NAME	ATTR_VALUE	ATTR_BASE_UOM_CODE	ATTR_BASE_UOM_VALUE
Capacitance		1 uF	Micro Farads
Capacitor case size	6mmx11mmlg		

In the data lens created by AutoBuild, the ATTR\_BASE\_UOM\_CODE is one of the term variants in the "[microfarad]" rule:

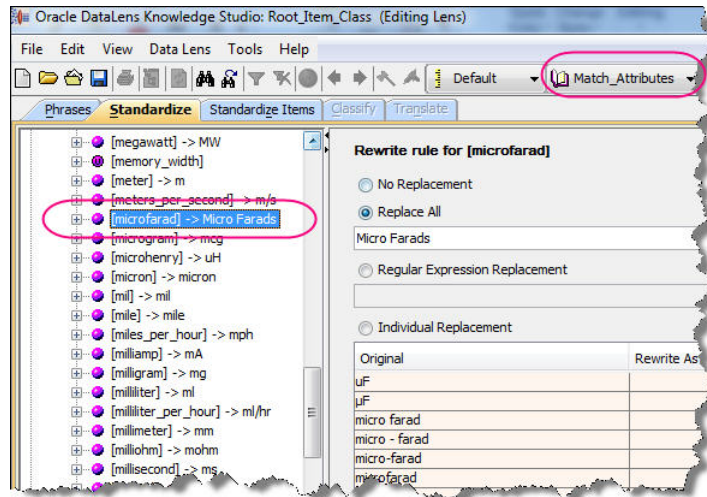


In the **Unit Conversion** tab, the microfarad (uF) is the target unit of measure for the "a\_capacitance" phrase rule:



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**:

ATTR_NAME	ATTR_VALUE	ATTR_BA	ATTR_VALUESSET	ATTR_MATCH
Capacitance		1 uF	Micro Farads	Y
Dielectric	ta		Dielectric_material	Y
Dielectric	mp		Dielectric_material	Y
Dielectric	ab		Dielectric_material	Y
Dielectric	al		Dielectric_material	Y
Dielectric	ce		Dielectric_material	Y
Tolerance		1 %	%	Y
Voltage		1 VLT	Volt	Y
Esr_type	HIGH		HIGH	N
Esr_type				N
Esr_value				N

Change the Match Ranking using the arrow buttons or with Drag and Drop

- Capacitance
- Dielectric
- Tolerance
- Voltage
- Capacitor\_case\_size
- Capacitor\_package\_size
- Esr\_type
- Esr\_value
- Load\_life
- Maximum\_temperature
- Mean\_time...

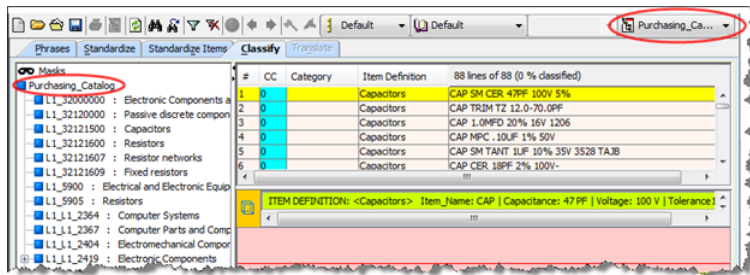
### Alternate Catalog Export

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:

Metadata	Description	Excel Column Header
Catalog Name	The name of the alternate catalog	CATALOG

Metadata	Description	Excel Column Header
Parent Category Code	The parent category code if not top-level category	PARENT_CATEGORY_ID
Parent Category Name	The parent category name if not top-level category	PARENT_CATEGORY_DESCRIPTION
Catalog Category Code	The category code.	CATEGORY_ID
Catalog Category Description	The name of the ICC	CATEGORY_DESCRIPTION
Language	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'	LANGUAGE

Given a Fusion PIMDH Catalog Name "Purchasing Catalog", the following classification type is added to the data lens:



### 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.

Metadata	Description	Excel Column Header
ICC Name	The item name	ITEM_CLASS
ICC Data	The item data	ITEM
ICC Description	The item description	DESCRIPTION

## 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 the **Load Meta-Data Files...** utility.
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**
    - **Open New / Updated 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 Reports check box is selected
  - **Use Alias for Id's** 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, the Oracle Enterprise Data Quality for Product Data login dialog box appears because the **Open New / Updated Data Lens** option was set.

13. Login to your Oracle DataLens Server.

The **Oracle Enterprise Data Quality for Product Data Launch Pad** appears, the Knowledge Studio is automatically started, your data lens is automatically selected to be opened, and you are prompted select a sample file. For more information, see *Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide*.

---

---

**Caution:** Do not click the **Oracle DataLens Knowledge Studio** button on the **Oracle Enterprise Data Quality for Product Data Launch Pad** because this causes the Knowledge Studio to stop responding so it does not open.

---

---

14. Examine your new data lens and refine the data recognition rules to add variants that may not have been automatically generated. For more information, see ["Refining Your New Data Lens"](#) on page 4-1.
15. Save your data lens, and then check it in with deployment to your Oracle DataLens Server.

## Updating an Existing Data Lens

An existing data lens can be updated using AutoBuild from an Item Class & 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**.

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:
  - 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:
  - **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**
  - **Open New / Updated 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, the Oracle Enterprise Data Quality for Product Data login dialog box appears because the **Open New / Updated Data Lens** option was set.

**11. Login to your Oracle DataLens Server.**

The **Oracle Enterprise Data Quality for Product Data Launch Pad** appears, the Knowledge Studio is automatically started, your data lens is automatically selected to be opened, and you are prompted select a sample file. For more information, see *Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide*.

---

---

**Caution:** Do not click the **Oracle DataLens Knowledge Studio** button on the **Oracle Enterprise Data Quality for Product Data Launch Pad** because this causes the Knowledge Studio to stop responding so it does not open.

---

---

**12. Examine your new data lens and refine the data recognition rules to add variants that may not have been automatically generated.**

**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.

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, the Oracle

Enterprise Data Quality for Product Data login dialog box appears because the **Open New / Updated Data Lens** option was set.

9. Login to your Oracle DataLens Server.

The **Oracle Enterprise Data Quality for Product Data Launch Pad** appears, the Knowledge Studio is automatically started, and you are prompted to select a sample file.

---

**Caution:** Do not click the **Oracle DataLens Knowledge Studio** button on the **Oracle Enterprise Data Quality for Product Data Launch Pad** because this causes the Knowledge Studio to stop responding so it does not open.

---

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 redisplay 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



---

---

## Continued DQ Improvement

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

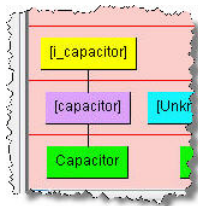
The data lens generated by Autobuild contains the structure you need for the PIM metadata synchronization. The Item Definitions, attributes, phrases, and terms have been created from the PIM export metadata; term rules and standardization rules can be modified in order to adequately handle the variability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

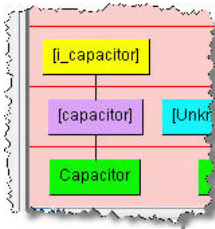
The following is an overview of recommendations for refining your data lens in the EDQP Knowledge Studio:

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

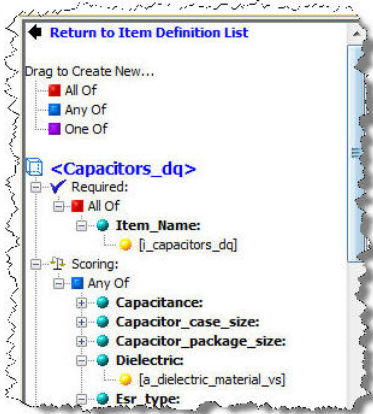


If the ICC name is not identical to the item name found in product data, use the following steps in Knowledge Studio:

- 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.

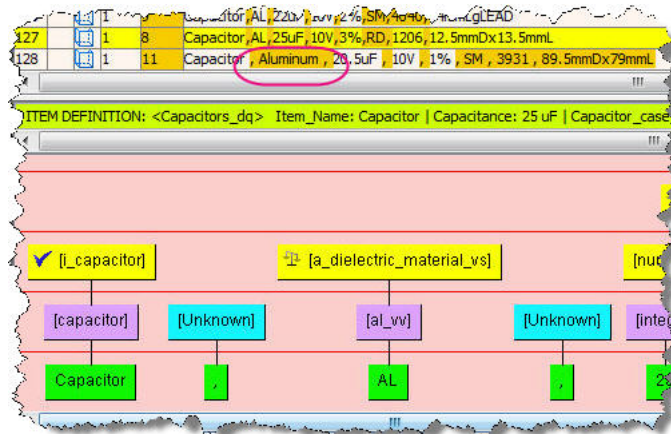


2. Review and resolve ambiguities.

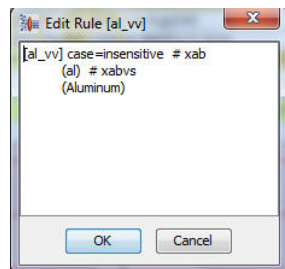
If there are any recognition or Item Definition ambiguities in your sample data, review and resolve them.

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 [a1\_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.

---

#### 6. Match rules.

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.

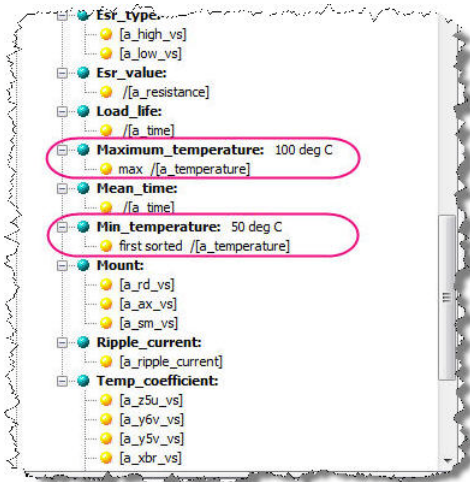
By default, the Item Definitions are inactive upon finishing the Autobuild process. Activate your Item Definitions before running a DSA that processes data through your data lens.

**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:

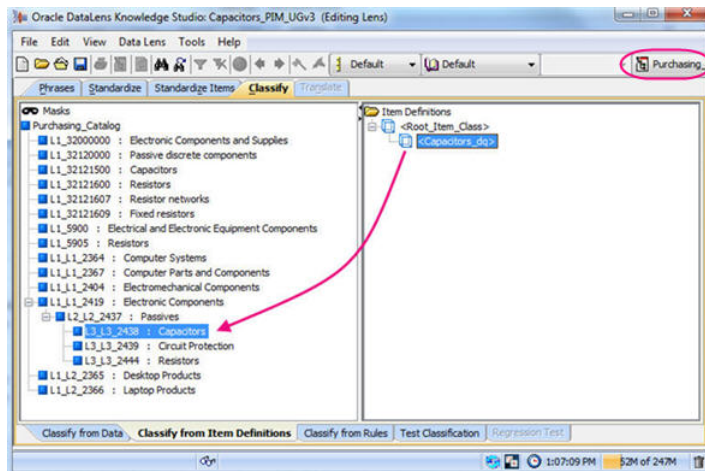
Capacitor,AL,22uF,10V,50 deg C, 100 deg C,1%,RD,.10S,10mmDx16mmL

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:

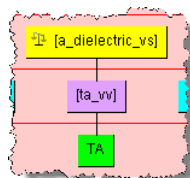
**Figure 4-1 Value Set Example Spreadsheet**

parent_catalog_group_id	parent_catalog_group	parent_catalog_group_id	catalog_group	attr_id	attr_name	attr_value	attr_uom	valueset
570	Passives	571	Capacitors	Capacitor_SpecsDielectric	Dielectric	AB		Dielectric
570	Passives	571	Capacitors	Capacitor_SpecsDielectric	Dielectric	AL		Dielectric
570	Passives	571	Capacitors	Capacitor_SpecsDielectric	Dielectric	CE		Dielectric
570	Passives	571	Capacitors	Capacitor_SpecsDielectric	Dielectric	MP		Dielectric
570	Passives	571	Capacitors	Capacitor_SpecsDielectric	Dielectric	TA		Dielectric
570	Capacitors	572	Capacitor Networks	Capacitor_SpecsShielded	Shielded	Yes		EGO_YES_NO
570	Capacitors	572	Capacitor Networks	Capacitor_SpecsShielded	Shielded	No		EGO_YES_NO
570	Passives	571	Capacitors	Capacitor_SpecsCAPACITANCE	CAPACITANCE	30	uF	
570	Passives	571	Capacitors	Capacitor_SpecsCAPACITANCE	CAPACITANCE	000225	uF	
570	Passives	571	Capacitors	Capacitor_SpecsCAPACITANCE	CAPACITANCE	47	uF	
570	Passives	571	Capacitors	Capacitor_SpecsCAPACITANCE	CAPACITANCE	20	uF	
570	Passives	571	Capacitors	Capacitor_SpecsCAPACITANCE	CAPACITANCE	20.89999	uF	
570	Passives	571	Capacitors	Capacitor_SpecsCAPACITANCE	CAPACITANCE	015	uF	

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:

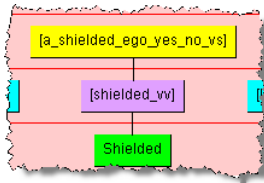
### Standard Value Sets

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:



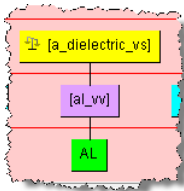
All variations of both yes and no are created in the data lens. For example, the variations of 'not shielded' would be:

'not shielded'  
 'not-shielded'  
 'not- shielded'  
 'not -shielded'  
 'not - shielded'  
 'notshielded'

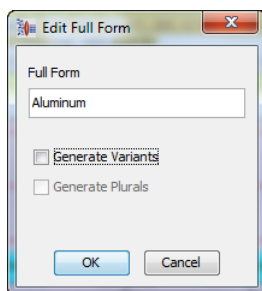
In the data lens, all value set phrase rule names are created with a '\_vs' appended to the attribute and '\_vv' appended to the term rule name so that they are easily identified in phrase structures.

## 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 terms to become reassociated with other phrases, it is necessary to review those attributes 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:

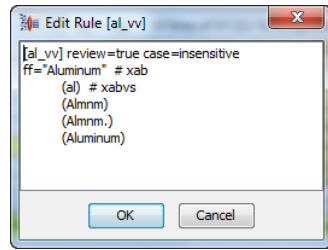


You would edit the 'al\_vv' term and add the 'Aluminum' full form (variants can be generated as well):

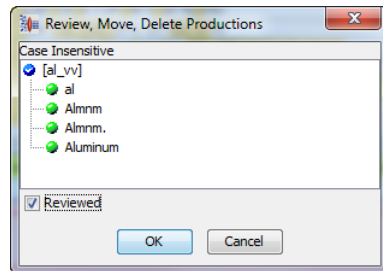




Then review the phrase rule to ensure that it was added properly:



Finally, review the productions of the 'a\_dielectric\_vs' phrase rule:




---

**Note:** Since AutoBuild ignores duplicate phrase productions when created from a value set, you should review all productions carefully.

---

## 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.

- 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:

E	ATTR_ID	ATTR_NAME	ATTR_VALUE	ATTR_BA	ATTR_VALUESSET
	Capacitors_AG.Temp_coefficient	Temp_coefficient	z6u		z6u
	Capacitors_AG.Temp_coefficient	Temp_coefficient	x8r		x8r
	Capacitors_AG.Temp_coefficient	Temp_coefficient	z5v		z5v
	Capacitors_AG.Temp_coefficient	Temp_coefficient	x7r		x7r
	Capacitors_AG.Capacitor_packag	Capacitor_package_size	ABC14		ABC14
	Capacitors_AG.Color_coding	Color Coding	blue		
	Capacitors_AG.Color_coding	Color Coding	red		
	Capacitors_AG.Color_coding	Color Coding	green		
	Capacitors_AG.Color_coding	Color Coding	yellow		

This additional information is represented in the **Semantic Knowledge Structure** report as follows. The attribute "Capacitor\_package\_size" has a new attribute value.

Attribute	Phrase/Term Associations
Item_Name	[i_capacitors_dq]
Capacitance	[/a_capacitance]
Capacitor_case_size	[a_cap_case_size_vs]
Capacitor_package_size	[a_cap_pkg_size_vs]
	[a_abc14_vs]
Dielectric	[a_dielectric_material_vs]
Tolerance	[/a_percent]
Voltage	[/a_voltage]
Esr_type	[a_high_vs]
	[a_low_vs]
Esr_value	[/a_resistance]
Load_life	[/a_time]
Maximum_temperature	[/a_temperature]
Mean_time	[/a_time]

This report also reports new term rules as follows:

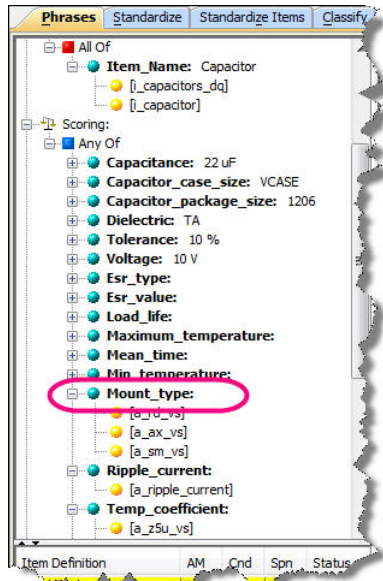
New	[a_color_coding]	[[blue]]
		[[red]]
		[[green]]
		[[yellow]]

### Renames

The attribute 'Mount' was renamed to 'Mount\_type' as in the following example:

Renamed	Mount_type	[a_rd_vs]
		[a_ax_vs]
		[a_sm_vs]

The renaming of the attribute in the data lens is performed automatically by AutoBuild:

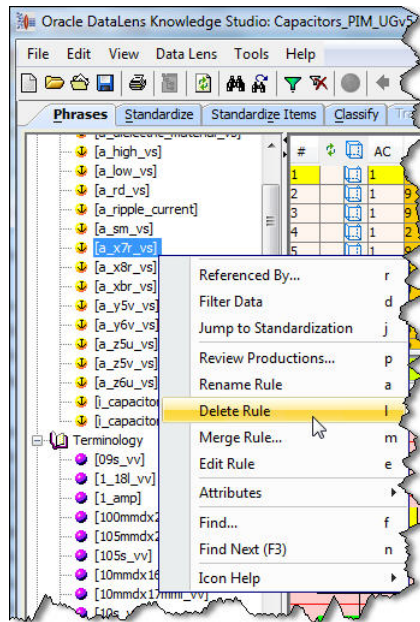


### Deletions

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.

DELETE	{a_x7r_vs}	{{x7r_vv}}
DELETE	{i_capacitor}	{{capacitor}}

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 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.

---

---

## Installing the Client Software

Enterprise DQ for Product uses Java Web Start to initially install and maintain the current version of the software on your client desktop. The process requires you to access the Oracle DataLens Server to initiate the connection and download the software.

The Enterprise DQ for Product client applications downloaded and installed using Java Web Start by browsing to the installation page for your Oracle DataLens Server as follows:

1. Ensure that you have the Java SE Runtime Environment (JRE) 6 Update 21 installed. You can download the JRE and obtain the installation instructions by browsing to:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

2. Start Microsoft Internet Explorer.
3. Initiate a connection and download the client software by browsing to:

`http://server:2229/datalens/datalens.`

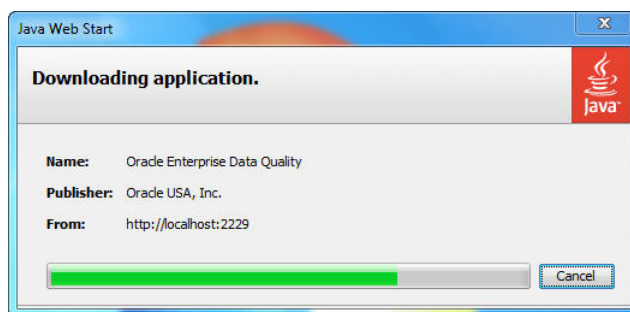
Where *server* is the hostname of the Oracle DataLens Server.

---

**Note:** If you have setup a different port number for your application server other than 2229, you must use that port number in the following URL when browsing to the Oracle DataLens Server to download the client applications.

---

The application download and verification begins.



---

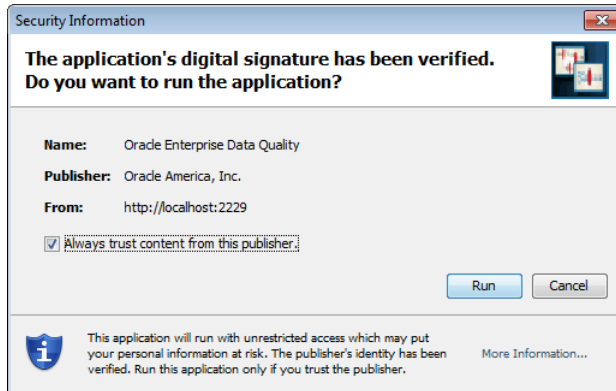
---

**Note:** If you receive a **File Download** message indicating that the . file is not associated with a program, you do not have the supported JRE installed. Click **Cancel** and return to Step

---

---

After the verification completes, the installation begins. Enterprise DQ for Product files are digitally signed by a trusted source so the following security warning is displayed:



**Tip:** To avoid this security dialog in the future, select the **Always trust content from this publisher** check box.

4. Click **Run** to continue and complete the installation.

The Oracle Enterprise Data Quality for Product Data log in dialog is displayed.

