



---

## *Agile PLM 9.3.2 and EDQP Integration White Paper*

*Part No. E38502-01*

---

*NOTE: This document is a supplement to the release Readme file and all other manuals, such as the "Capacity Planning Guide". It is intended to be an overview of the other documentation and does not replace the need to read and understand those documents. Errors contained in this document are not commitments to provide features.*

## Table of Contents

1	Overview .....	2
2	Configuring EDQP with the Agile PLM Database .....	3
2.1	Getting Started .....	3
3	Installing EDQP templates and configuring the Agile PLM integration .....	4
3.1	Creating the PLM_CONNECTOR Database Connection .....	4
3.2	Installing EDQP Agile PLM sample files .....	6
3.3	Updating Excel Services.....	7
3.4	Enterprise DQ for Product DSA Sample Files Overview .....	8
4	Enriching the Agile PLM Dataset with EDQP .....	9
4.1	Validating Attributes .....	9
4.2	Create Semantic Model Input .....	10
4.3	Using Services for Excel AutoBuild to Create a Data Lens .....	13
4.4	Reviewing and Refining the Data Lens .....	15
4.5	Cleansing and Enriching Agile Production Data .....	17
4.5.1	Create Production Batch.....	17
4.5.2	Cleanse Production Batch .....	18
4.6	Importing Cleansed Data back into Agile .....	20
4.6.1	Basic Import Process.....	20
4.6.2	Additional process considerations.....	23

## 1 Overview

---

Poor data quality creates issues for an Agile Product Lifecycle Management (PLM) system. People enter information into Agile PLM in an inconsistent manner with incorrect and missing information. For example from an NEMI study, 40-80% of BOMs arrive at Procurement/Manufacturing with 'problems' including:

- Inconsistent content (missing and cryptic information)
- Inconsistent formats (including languages)
- Invalid or missing supplier information
- Invalid or missing part information
- Obsolete parts

Resolution of these inconsistencies can add days or weeks to new product introduction (and creates additional cost). The solution is to enforce consistent standards on all parts and BOMs using an integrated Data Quality solution. Since 70% of product costs are locked-in early, it's important to deliver integrated data quality from the beginning of the process.

In Agile PLM 9.3.2, Enterprise Data Quality for Product (EDQP) has access to Agile PLM administrative information, as well as the entire Item data set through direct database access. This access allows EDQP to be integrated to Agile PLM in a high performance manner that enables use cases for data validation and enrichment. This integration enables processes to manage the cleanliness of the Agile PLM enterprise product record.

## 2 Configuring EDQP with the Agile PLM Database

---

The integration of the EDQP solution with Agile PLM 9.3.2 provides an integrated set of capabilities to categorize, standardize, match, govern, validate, correct, and repurpose both existing Agile PLM items and product data being introduced from any source system(s). It provides a set of ready-to-use integration templates, in addition to a robust set of interface functions and procedures that can be leveraged to create customized process flows. These process flows cleanse, standardize, and extract attributes for existing Agile PLM data and provide the foundation for additional data quality capabilities.

The included templates allow a quick path to performing the following functions:

- Metadata Synchronization Template:
  - Automatically generate semantic models from the subclass metadata, including page three attributes with their associated valid values and UOMs if available.
- Create Production Batch to cleanse and revalidate existing items in your Agile PLM system.
- Run cleansing process to cleanse data, standardized attributes, and create standardized description for Agile Production batches.
- Create a standard import Excel worksheet ready for import into Agile PLM.

The Agile PLM integration includes the necessary tables to enable customization of the EDQP process to perform the following functions:

- Develop a full remediation process to enable business users to correct and complete item records.
- Cleanse, extract, and standardize external data for import into Agile PLM.
- Create Semantic Key Cache for duplicate detection and resolution for existing items and externally loaded items,
- Cleanse and standardize Manufacturer Names and Manufacturer Part Numbers.

### 2.1 Getting Started

---

You must ensure that you have completed the following actions before using the Agile PLM Connector:

- Install the Agile PLM 9.3.2 server (including the latest patch sets)
- Ensure that the Oracle database support the Agile PLM 9.3.2 server is setup correctly using the [Agile Product Lifecycle Management Database Installation Guide v9.3.2](#).
  - Enable Oracle JVM on the Agile PLM database.
    - The connector uses Java stored procedures, pdqplm\_plsql.jar, included with Agile database utilities.
  - Run the agile9pdq\_setup file

- Install and configure an EDQP 11g Oracle DataLens Server. For instructions, see [Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Installation Guide](#).
- Install and configure the EDQP Services for Excel 11g. For instructions, see [Oracle Enterprise Data Quality for Product Data Services for Excel Reference Guide](#).

### 3 Installing EDQP templates and configuring the Agile PLM integration

---

The following section describes the configuration and sample template installation for the EDQP and Agile PLM integration.

#### 3.1 Creating the PLM\_CONNECTOR Database Connection

---

You must create a database connection for use by the EDQP DSAs to process data. This database connection is created in the EDQP Oracle DataLens Administration Server using the Oracle DataLens Administration Server web page. For more information, see [Oracle Enterprise Data Quality for Product Data Oracle DataLens Server Administration Guide](#).

1. Browse to the Oracle DataLens Administration Server web page:  
`http://hostname:port/datalens`  
where *hostname* is the DNS name or IP address of the Administration Server and *port* is the listen port on which the Administration Server is listening for requests (port 2229 by default).
2. Log in.
3. Click the **Administration** tab.
4. From the **Configuration** section of the menu, click **Database Connections**.
5. Click the **New Connection** button.

**Create DB Connection**

\* Name

\* Description

Area Development

\* Connection Type

\* Database Type

\* Connection

\* Driver

Test String

\* Db User

DB Password

Initial Db Connections

Max Connections

DB Timeout (Minutes)

DB Batch Size

6. In the Connection Name field, enter **PLM\_CONNECTOR** and Description.  
The Connection Name must be this exact name or the process will not work correctly; the DSAs rely on the presence of this exactly named connection.
7. Select **Standard** as the **Connection Type**. The form is then populated for you with default information.
8. Select **Oracle** as the Database Type. The form is then populated for you with default information.
9. Complete the form:
  - a. Enter a description for this database connection.
  - b. In the **Connection** field, edit the machine name, Oracle Database Port, and Database SID to match your environment.  
In the following example, these values are set to:  
scl34099.us.oracle.com:1521:tf932b47  
The Oracle Database Port default value is 1521.
  - c. Enter the Db User name.
  - d. Enter the DB User password.

View Database Connection for - APLM\_CONNECTOR

Development

\* Description: APLM\_CONNECTOR

\* Connection: jdbc:oracle:thin:@scl34099.us.oracle.com:1521:tf932b47

\* Driver: oracle.jdbc.driver.OracleDriver

Connection Type: Standard

Test String: SELECT \* FROM dual

\* Db User: b80sales

DB Password: .....

Initial Db Connections: 5

Max Connections: 40

DB Timeout (Minutes): 60

DB Batch Size: 15000

Save Test

10. Click the **Test** button to verify that your EDQP Named Development Database Connection is working properly. The following message should be displayed:

Test Output Successful

If the connection does not test successfully, review and revise the options for accuracy, and then retest.

11. Click **Save**.

The connection information is saved and it is added to the list of database connections to which you are returned.

## 3.2 Installing EDQP Agile PLM sample files

---

The Agile PLM Connector Application Server sample files are a collection of DSAs that are delivered with EDQP Agile PLM Connector Patch. You can install them on your Oracle DataLens Server as an example of how you can process your data using the Agile PLM sample data. This sample includes a set of DSAs, each of which is described in this section.

Install the Agile PLM Connector application server sample 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 contents of the patch to the `autodeploy` directory located on your Oracle DataLens Administration Server WebLogic `dls_domain`. The DSAs end with a `.XML` file extension and should be placed in the `dsa` subdirectory while the data lenses end with a `.project` file extension and should be placed in the `lens` subdirectory. By default, these directories are:

On Linux and UNIX:

```
MW_HOME/user_projects/domains/dls_domain/opdq/autodeploy/dsa
MW_HOME/user_projects/domains/dls_domain/opdq/autodeploy/lens
```

On Windows:

```
MW_HOME\user_projects\domains\dls_domain\opdq\autodeploy\dsa
MW_HOME\user_projects\domains\dls_domain\opdq\autodeploy\lens
```

**Note:**

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

This automatically deploys (autodeploys) the DSA sample files to your Oracle DataLens Server. The Oracle DataLens Administration Server polls the `autodeploy` directory every 10 minutes and attempts to autodeploy the DSA sample files that are placed in the `autodeploy` parent directory.

3. Log out of the Oracle DataLens Administration Server system.
4. Wait 10 minutes to ensure that the server samples 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 APLM\_NAME DSAs have been autodeployed.

### 3.3 Updating Excel Services

---

The EDQP Services for Excel has the capability to normalize the view of attribute output name / value column pairs by changing them into a single column with the attribute name as the header value suitable for import into the Agile PLM system. To turn on this capability, the configuration file, `dlsforexcel_cfg.xml` file must be updated with one line.

1. Close any open Excel workbooks.
2. Navigate to the following directory `%AppData%\DataLens\config\`.
3. Edit the `dlsforexcel_cfg.xml` file.
4. Add the following line at the end of the file, and then save it:

```
<data
property="app.ui.feature.groupintoworksheets.mergeattribute
namevalue" value="true" />
```

The file should look similar to the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<dlsforexcel version="1.0" datetime="11/8/2012 1:53:19 PM">
  <properties>
    <data property="app.locale" value="en_us" />
    <data property="app.servergroup" value="Admin" />
    <data property="app.transformserver" value="localhost" />
    <data property="app.transformserver.port" value="2229" />
    <data property="app.username" value="dlsadmin" />
    <data property="app.password" value="xxxxxxx" />
    <data property="app.adminserver" value="localhost" />
    <data property="app.adminserver.port" value="2229" />
    <data property="app.rememberpassword" value="True" />
    <data property="app.usehttps" value="False" />
    <data property="app.session.folder.path"
value="C:\Users\username\Desktop" />
    <data
property="app.ui.feature.groupintoworksheets.mergeattributenamevalue"
value="true" />
  </properties>
</dlsforexcel>
```

### 3.4 Enterprise DQ for Product DSA Sample Files Overview

---

There are three main categories of samples DSAs that illustrate the overall integration capabilities between the EDQP and Agile PLM.

- Metadata Import that creates a Semantic Model from Agile PLM metadata extract.
- Creating a new batch from a query of production data.
- Cleansing the batch of production data to produce an importable spreadsheet for import into the Agile PLM server.

#### **APLM\_CREATE\_SEMANTIC\_MODEL**

This DSA retrieves the semantic model metadata for a given sub-class from the Agile PLM system. EDQP AutoBuild (an application within Services for Excel) uses this semantic model metadata to build a data lens that is used as a starting point for the cleansing and enrichment process. This DSA is typically run using the Services for Excel to create the data needed by the AutoBuild application. This data includes attributes and sample descriptions.

#### **APLM\_CREATE\_PRODUCTION\_BATCH**

This DSA is run to create a new batch of data from the Agile PLM Production tables. This batch is then processed by the Oracle DataLens Server and imported back into the Agile PLM system to update the item records with the cleansed and enriched data.

#### **APLM\_CLEANSE\_PRODUCTION\_BATCH**

This DSA is run as the main DSA. It processes an Agile PLM Production batch to cleanse, standardize and extract attributes from the description and the original concatenated attribute values.

#### **APLM\_Capacitors**

This data lens is used in the APLM\_CLEANSE\_PRODUCTION\_BATCH DSA to cleanse, standardize and extract attributes from the description and the original concatenated attribute values.

#### **Default Entries in DSAs**

All DSAs shipped as part of the EDQP Agile PLM Connector have default entries that allow quick configuration.

#### **Default Database Connection**

The default database connection name for all DSAs is 'APLM\_CONNECTOR'. This database connection points to a fictitious server name with a fictitious user and password that must be configured as part of the initial server configuration, see [3.1 Creating the PLM CONNECTOR Database Connection](#).



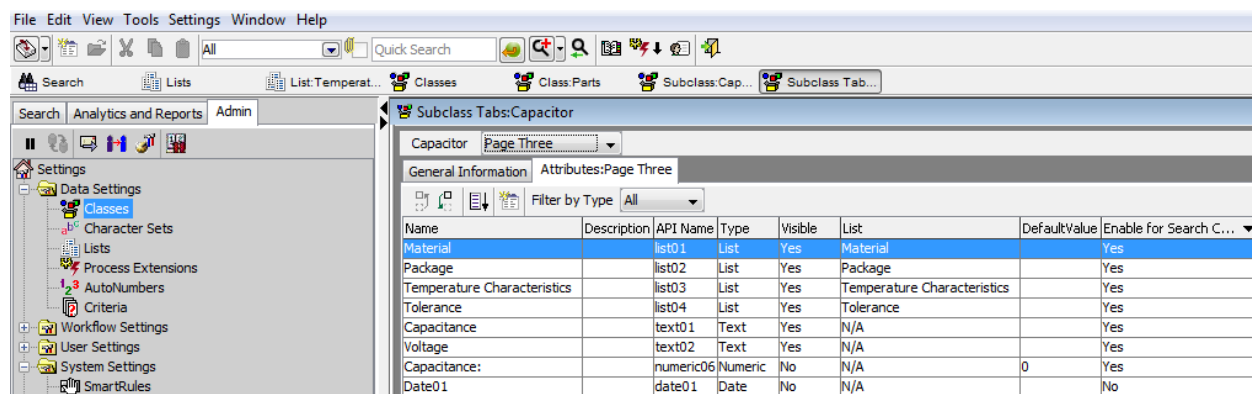
## 4 Enriching the Agile PLM Dataset with EDQP

Governance is important to any data quality initiative where standards are put into place that considers the organizational needs to establish the necessary attributes for the business process. Although EDQP can apply standards to any type of data value, it is especially important for form, fit and function attributes that appear on page 3. The Agile PLM system has Page 3 attribute definitions per category (sub class).

### 4.1 Validating Attributes

The first step in the data quality process is to analyze the existing set of parts then to define and set up the Page 3 attributes to enable the standardization, extraction and enrichment of those attributes from the existing part description and attribute values.

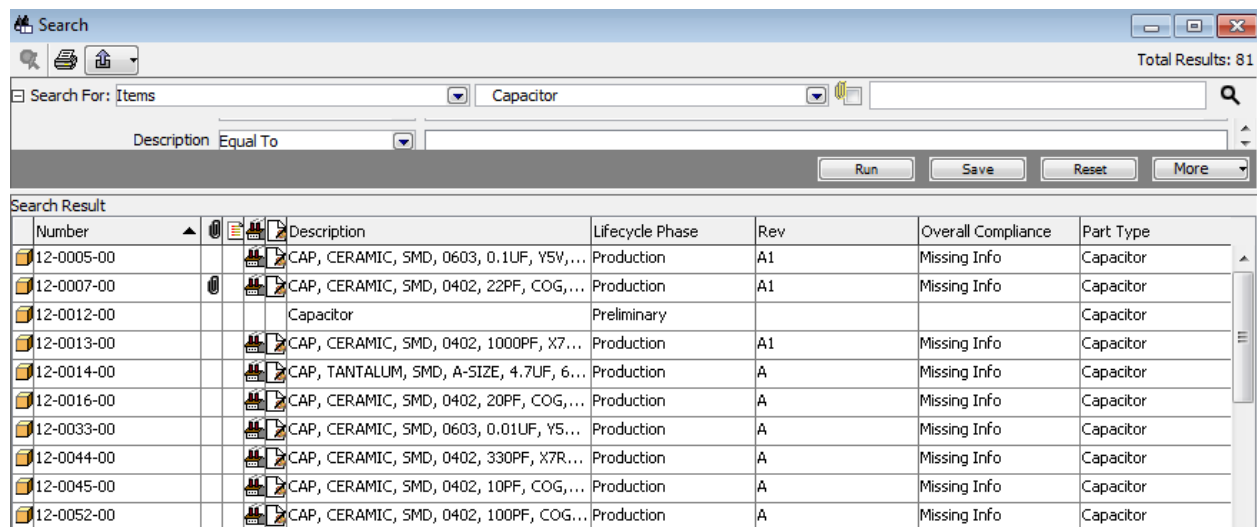
Following is an example of the setup for the Page 3 attributes for a Capacitor in a demonstration Agile PLM system:



Name	Description	API Name	Type	Visible	List	DefaultValue	Enable for Search C...
Material		list01	List	Yes	Material		Yes
Package		list02	List	Yes	Package		Yes
Temperature Characteristics		list03	List	Yes	Temperature Characteristics		Yes
Tolerance		list04	List	Yes	Tolerance		Yes
Capacitance		text01	Text	Yes	N/A		Yes
Voltage		text02	Text	Yes	N/A		Yes
Capacitance:		numeric06	Numeric	No	N/A	0	Yes
Date01		date01	Date	No	N/A		No

Notice that there are six attributes: Material, Package, Temperature Characteristics, Tolerance, Capacitance and Voltage. You should validate that they are setup correctly and represent the entire set of attributes required for that part.

There are currently 81 Capacitors in this demo Agile PLM system. The Agile EDQP connector retrieves these items and uses them as sample data when creating a Semantic Model.



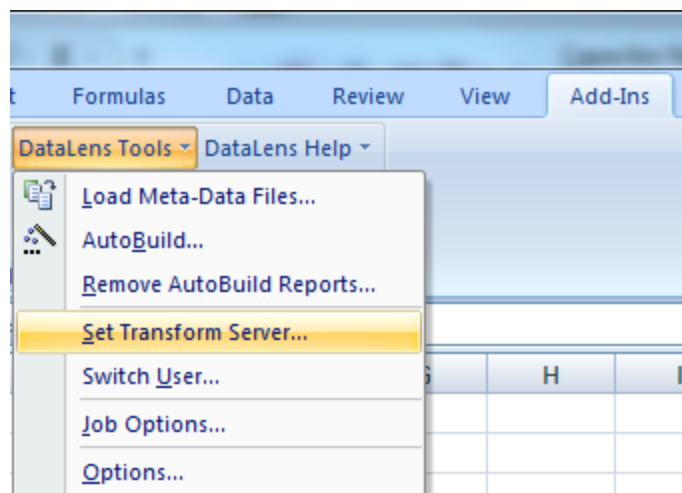
Number	Description	Lifecycle Phase	Rev	Overall Compliance	Part Type
12-0005-00	CAP, CERAMIC, SMD, 0603, 0.1UF, Y5V,...	Production	A1	Missing Info	Capacitor
12-0007-00	CAP, CERAMIC, SMD, 0402, 22PF, COG,...	Production	A1	Missing Info	Capacitor
12-0012-00	Capacitor	Preliminary			Capacitor
12-0013-00	CAP, CERAMIC, SMD, 0402, 1000PF, X7R,...	Production	A1	Missing Info	Capacitor
12-0014-00	CAP, TANTALUM, SMD, A-SIZE, 4.7UF, 6V,...	Production	A	Missing Info	Capacitor
12-0016-00	CAP, CERAMIC, SMD, 0402, 20PF, COG,...	Production	A	Missing Info	Capacitor
12-0033-00	CAP, CERAMIC, SMD, 0603, 0.01UF, Y5V,...	Production	A	Missing Info	Capacitor
12-0044-00	CAP, CERAMIC, SMD, 0402, 330PF, X7R,...	Production	A	Missing Info	Capacitor
12-0045-00	CAP, CERAMIC, SMD, 0402, 10PF, COG,...	Production	A	Missing Info	Capacitor
12-0052-00	CAP, CERAMIC, SMD, 0402, 100PF, COG,...	Production	A	Missing Info	Capacitor

## 4.2 Create Semantic Model Input

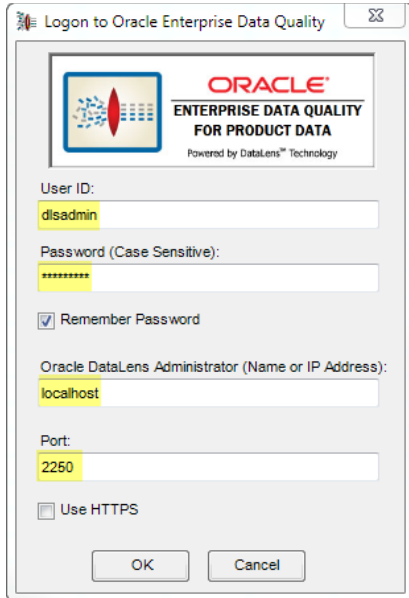
Once the page three attributes are validated and setup for all of the relevant subclasses, you run the metadata export process called Create Semantic model to download the structure and data into an Excel worksheet using the Agile EDQP Connector integration.

In Excel, you use the Services for Excel application to run the Create Semantic Model process as follows:

1. Start Excel and open a new workbook.
2. Select the **Add-Ins** tab.
3. Select **Set Transform Server...** from the DataLens Tools menu to connect to the EDQP DataLens server.

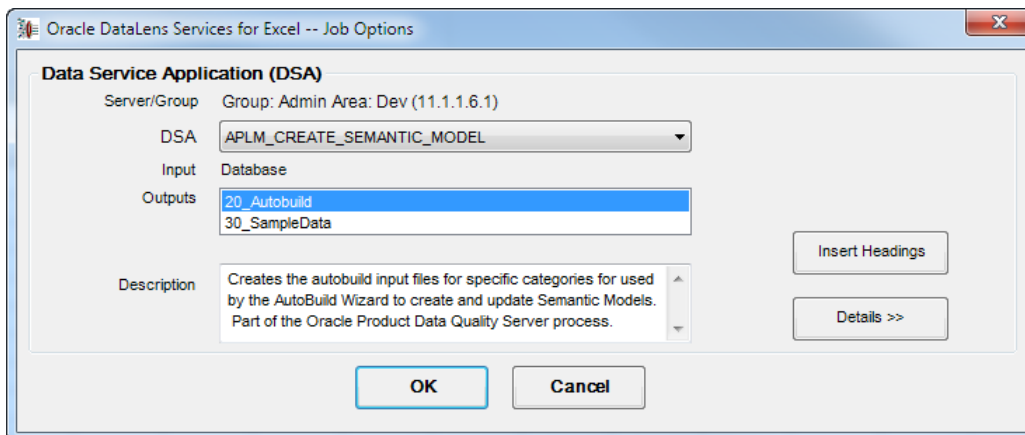


1. Log in to the EDQP server with the correct User ID, Password, Server Name and Port.



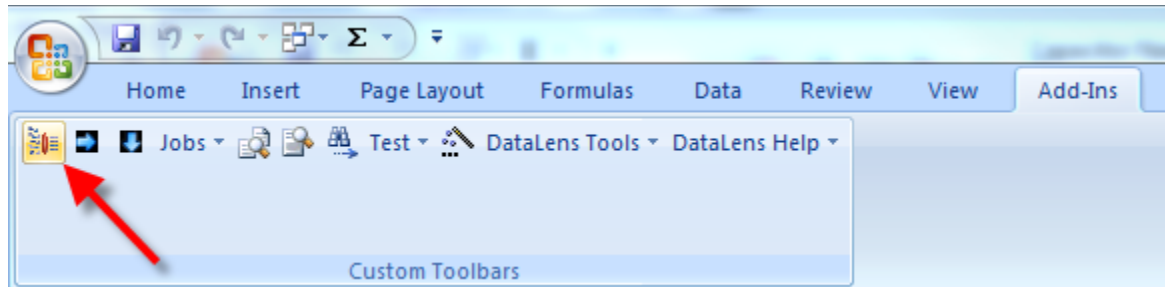
The screenshot shows a login window titled "Ligon to Oracle Enterprise Data Quality". It features the Oracle logo and the text "ENTERPRISE DATA QUALITY FOR PRODUCT DATA" and "Powered by DataLens™ Technology". The form includes fields for "User ID:" (containing "disadmin"), "Password (Case Sensitive):" (masked with asterisks), a checked "Remember Password" checkbox, "Oracle DataLens Administrator (Name or IP Address):" (containing "localhost"), and "Port:" (containing "2250"). There is an unchecked "Use HTTPS" checkbox and "OK" and "Cancel" buttons at the bottom.

2. Accept the default of the Server Group, Admin, by clicking **OK**.
3. Select **Job Options...** from the DataLens Tools menu to select the DSA to process the job on the EDQP DataLens server.
4. Select **APLM\_CREATE\_SEMANTIC\_MODEL** from the DSA list and click **OK**.

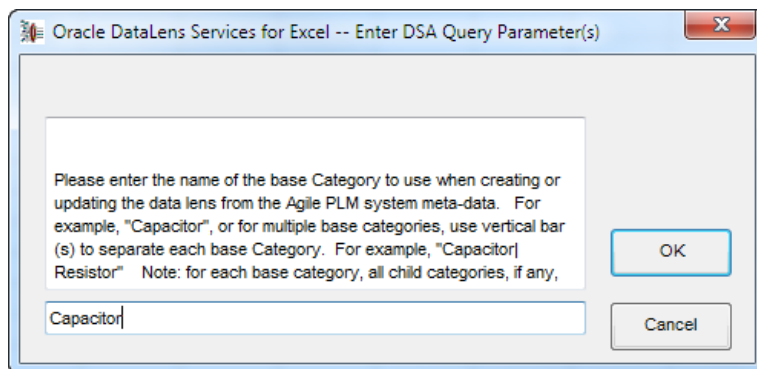


The screenshot shows a dialog box titled "Oracle DataLens Services for Excel -- Job Options". It contains a "Data Service Application (DSA)" section with the following fields: "Server/Group" (Group: Admin Area: Dev (11.1.1.6.1)), "DSA" (a dropdown menu showing "APLM\_CREATE\_SEMANTIC\_MODEL"), "Input" (Database), and "Outputs" (a list box containing "20\_Autobuild" and "30\_SampleData"). A "Description" field contains the text: "Creates the autobuild input files for specific categories for used by the AutoBuild Wizard to create and update Semantic Models. Part of the Oracle Product Data Quality Server process." There are "Insert Headings" and "Details >>" buttons on the right, and "OK" and "Cancel" buttons at the bottom.

1. Click the run button to initiate the job processing.



2. Enter the subclass (category) name(s) with a pipe separator between them if more than one of the item (for example **Capacitor**) to configure the DSA to retrieve the correct subclass names from the Agile PLM system using the Agile EDQP Connector then click **OK**.



3. After the results are retrieved, click **OK** to return to the worksheet to see the results.

	A	B	C	D	E	F	G	H	I
1	parent_catalog_group_id	parent_catalog_group	catalog_group_id	catalog_group	attr_id	attr_name	attr_value	attr_uom	valueset
2	0		10000	Part Type	1081				
3	10000	Part Type	2467580	Capacitor					
4	10000	Part Type	2467580	Capacitor	1575	Capacitance	1000pF		
5	10000	Part Type	2467580	Capacitor	1575	Capacitance	.33uF		
6	10000	Part Type	2467580	Capacitor	1575	Capacitance	.01uF		
7	10000	Part Type	2467580	Capacitor	1575	Capacitance	.018uF		
8	10000	Part Type	2467580	Capacitor	1575	Capacitance	1.0uF		
9	10000	Part Type	2467580	Capacitor	1575	Capacitance	.1uF		
10	10000	Part Type	2467580	Capacitor	1575	Capacitance	100pF		
11	10000	Part Type	2467580	Capacitor	1575	Capacitance	4.7uF		
12	10000	Part Type	2467580	Capacitor	1575	Capacitance	15uF		
13	10000	Part Type	2467580	Capacitor	1575	Capacitance	20pF		
14	10000	Part Type	2467580	Capacitor	1575	Capacitance	33uF		
15	10000	Part Type	2467580	Capacitor	1576	Voltage	40 - 90 F		
16	10000	Part Type	2467580	Capacitor	1576	Voltage	22.1K		
17	10000	Part Type	2467580	Capacitor	1576	Voltage	30 - 100 F		

The data is in a format that is ready for the AutoBuild process to run to automatically create the data lens that contains the data quality rules.

### 4.3 Using Services for Excel AutoBuild to Create a Data Lens

---

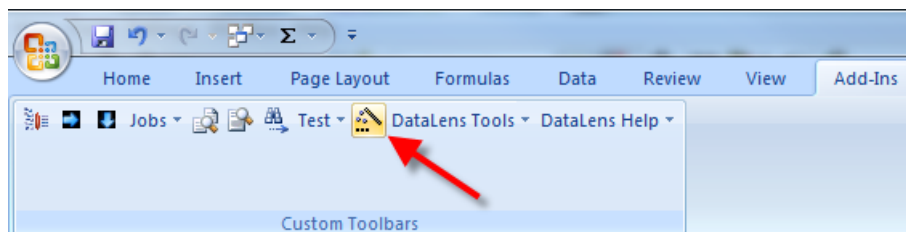
AutoBuild constructs a data lens by examining the available product metadata and sample data. Given sufficient information, AutoBuild can accomplish the following.

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

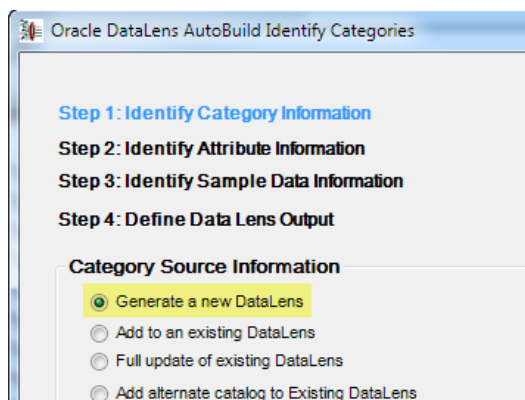
AutoBuild offers a familiar, easy-to-use graphical wizard interface that guides you through the process from start to finish.

The Autobuild process extracts data from the APLM schema and uses it to create a data lens that will cleanse, standardize, classify and extract attributes from the Agile PLM data.

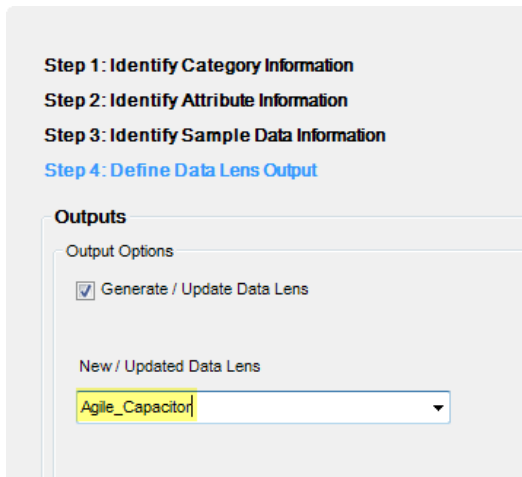
1. Click the AutoBuild button to initiate the AutoBuild processing to create the data lens from the Agile PLM data.



2. Select **Generate a new DataLens** radio button in Step 1 of the AutoBuild wizard. It automatically populates all of the values for this step and the remainder of the wizard steps because the columns are a special format that AutoBuild is designed to recognize this is built using a connector template.



3. Click on **Next** twice to move to the last step (Step 4).
4. Name the data lens (for example **Agile\_Capacitor**), and then click **Finish** to create the data lens.



Step 1: Identify Category Information  
Step 2: Identify Attribute Information  
Step 3: Identify Sample Data Information  
Step 4: Define Data Lens Output

**Outputs**

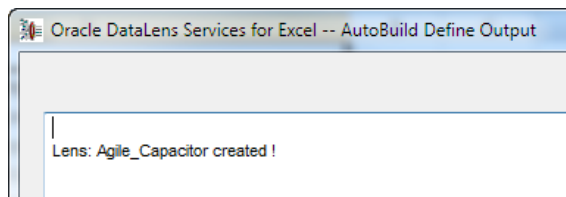
Output Options

☒ Generate / Update Data Lens

New / Updated Data Lens

Agile\_Capacitor

The process runs and a message is displayed that it successfully created the data lens.



5. Click **OK** to exit the dialogue. Your initial AutoBuild lens is complete and you are ready to review and refine the data lens.

## 4.4 Reviewing and Refining the Data Lens

6. Review the data lens to confirm proper creation. The data lens should have the correct Item Definitions, Attributes, Phrases, and Terminology as defined by the AutoBuild process.

You should review the data lens and add any undefined terminology to complete the process. For example, the following terms "CERAMIC" AND "SMD" not being recognized by the "Capacitor" Item Definition thus they are "Unknown".

The screenshot displays the Oracle DataLens Knowledge Studio interface for editing a lens named 'Agile\_Capacitor'. The main window shows a list of 74 lines of data, with 40% recognized. The data is organized into columns: #, AC, UC, and a description. The description column contains various capacitor specifications, including 'CAP, CERAMIC, SMD, 0603, 0.1UF, Y5V, 25V, -20/+80%', 'CAP, CERAMIC, SMD, 0402, 22PF, COG, 50V, 5%', 'Capacitor', 'CAP, CERAMIC, SMD, 0402, 1000PF, X7R, 50V, 10%', 'CAP, TANTALUM, SMD, A-SIZE, 4.7UF, 6.3V, 20%', 'CAP, CERAMIC, SMD, 0402, 20PF, COG, 50V, 5%', and 'CAP, CERAMIC, SMD, 0603, 0.01UF, Y5V, 50V, -20/+80%'. Below the list, the 'ITEM DEFINITION: <Capacitor>' is shown, detailing the item name, capacitance, voltage, and package. The definition is visualized as a tree structure with nodes for '[\_capacitor]', '[a\_package]', '[num]', '[capacitor]', '[Unknown]', '[Unknown]', '[Unknown]', '[Unknown]', '[Unknown]', '[0603]', '[Unknown]', and '[dec]'. The tree structure shows the hierarchy of the item definition, with nodes for '[\_capacitor]', '[a\_package]', '[num]', '[capacitor]', '[Unknown]', '[Unknown]', '[Unknown]', '[Unknown]', '[Unknown]', '[0603]', '[Unknown]', and '[dec]'. The nodes are color-coded: green for known terms (CAP, CERAMIC, SMD, 0603, 0.1) and blue for unknown terms ([Unknown]).

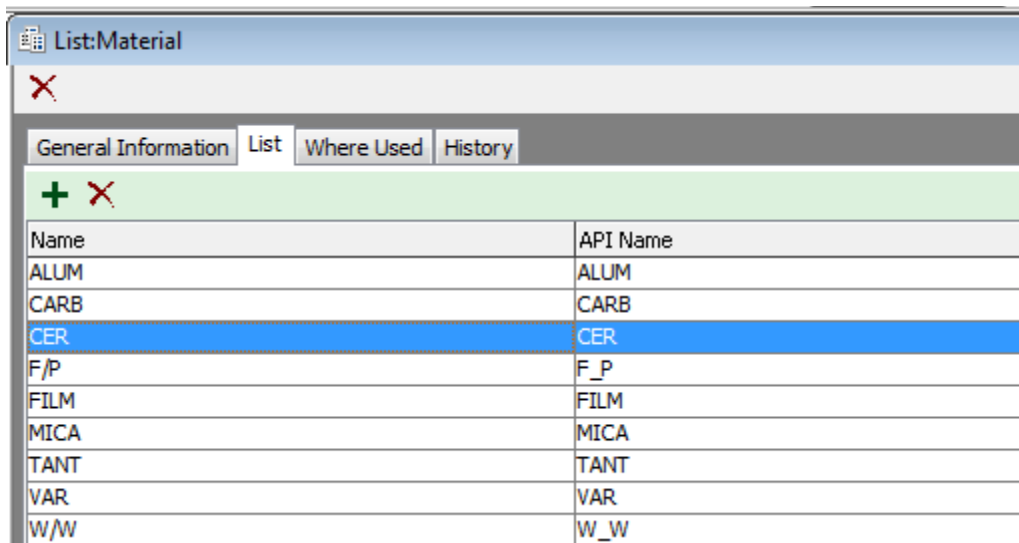
#	AC	UC	Description
1	1	11	CAP, CERAMIC, SMD, 0603, 0.1UF, Y5V, 25V, -20/+80%
2	1	10	CAP, CERAMIC, SMD, 0402, 22PF, COG, 50V, 5%
3	1	0	Capacitor
4	1	9	CAP, CERAMIC, SMD, 0402, 1000PF, X7R, 50V, 10%
5	1	9	CAP, TANTALUM, SMD, A-SIZE, 4.7UF, 6.3V, 20%
6	1	10	CAP, CERAMIC, SMD, 0402, 20PF, COG, 50V, 5%
7	1	11	CAP, CERAMIC, SMD, 0603, 0.01UF, Y5V, 50V, -20/+80%

ITEM DEFINITION: <Capacitor> Item\_Name: CAP | Capacitance: 0.1 UF | Voltage: 25 V | Package: 0603

Tree structure:

- [\_capacitor]
  - [capacitor]
    - CAP
  - [a\_package]
    - [Unknown]
    - [Unknown]
    - [Unknown]
    - [Unknown]
    - [Unknown]
    - [Unknown]
    - [0603]
    - [Unknown]
    - [dec]

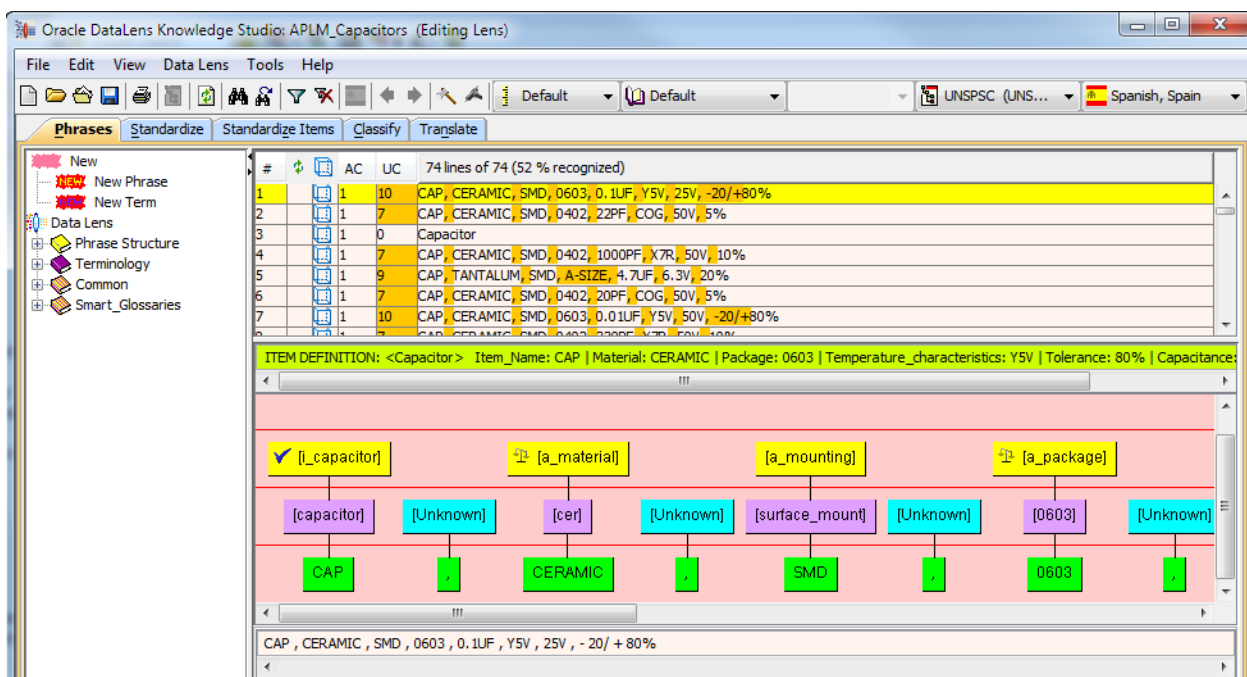
“CERAMIC” is not recognized as a variant of Agile standard “CER”. You would have to update the data lens and add the variant to the term “CER”.



Name	API Name
ALUM	ALUM
CARB	CARB
CER	CER
F/P	F_P
FILM	FILM
MICA	MICA
TANT	TANT
VAR	VAR
W/W	W_W

In the case of “SMD”, it was not setup on page 3 originally. If you wanted to include this attribute, you would go back to the Agile PLM to setup the page 3 attributes for Mounting (SMD) for Surface Mount then return to step [4.2](#) and either recreate or update the data lens using AutoBuild to reflect the changes.

After making the adjustments, the data lens will recognize the terms. The example data lens, APLM\_Capacitors, included with the templates, is a finalized data lens as in the following.



Oracle DataLens Knowledge Studio: APLM\_Capacitors (Editing Lens)

File Edit View Data Lens Tools Help

Phrases Standardize Standardize Items Classify Translate

74 lines of 74 (52 % recognized)

#	AC	UC	74 lines of 74 (52 % recognized)
1	1	10	CAP, CERAMIC, SMD, 0603, 0.1UF, Y5V, 25V, -20/+80%
2	1	7	CAP, CERAMIC, SMD, 0402, 22PF, COG, 50V, 5%
3	1	0	Capacitor
4	1	7	CAP, CERAMIC, SMD, 0402, 1000PF, X7R, 50V, 10%
5	1	9	CAP, TANTALUM, SMD, A-SIZE, 4.7UF, 6.3V, 20%
6	1	7	CAP, CERAMIC, SMD, 0402, 20PF, COG, 50V, 5%
7	1	10	CAP, CERAMIC, SMD, 0603, 0.01UF, Y5V, 50V, -20/+80%

ITEM DEFINITION: <Capacitor> Item\_Name: CAP | Material: CERAMIC | Package: 0603 | Temperature\_characteristics: Y5V | Tolerance: 80% | Capacitance:

[l\_capacitor] [a\_material] [a\_mounting] [a\_package]

[capacitor] [Unknown] [cer] [Unknown] [surface\_mount] [Unknown] [0603] [Unknown]

CAP, CERAMIC, SMD, 0603, 0.1UF, Y5V, 25V, -20/+80%



The update of data lens to include updates to new knowledge rules and the best practices of creating knowledge are outside the scope of this white paper. For more information on how to work with data lenses, please see [Oracle Enterprise Data Quality for Product Data Knowledge Studio Reference Guide](#).

## 4.5 Cleansing and Enriching Agile Production Data

---

Once you have created the data lens using one or more subclasses, you can now use the data lens to cleanse, extract, and standardize Agile PLM data.

### 4.5.1 Create Production Batch

---

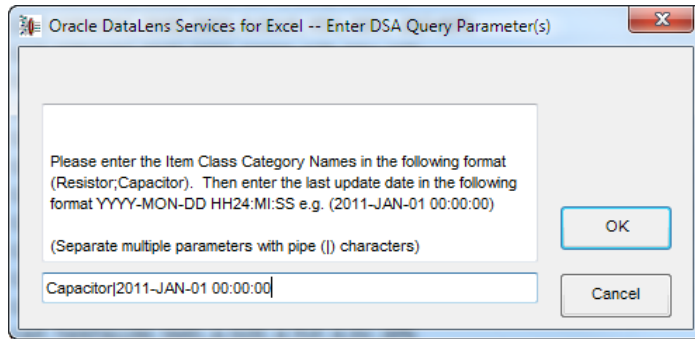
The first step in the process is to create a production batch to process. The production batch runs database queries to retrieve data for particular categories (subclasses) and combines them into a batch for processing by the EDQP software. Since the Agile PLM categories are flat, the connector allows multiple categories to be retrieved in a single batch using a semi-colon (;) as a separator. For example “Capacitor;Resistor”.

1. Start Excel and go to the Add-Ins tab.
2. Select the Transform server as shown in step [4.2](#).
3. Select **Job Options...** from the DataLens Tools Menu to select the DSA to process the job on the EDQP Oracle DataLens Server.
4. Select **APLM\_CREATE\_PRODUCTION\_BATCH** from the DSA list, and then click **OK**.
5. Click the run button to initiate the job processing.
6. Enter the subclass (category) name(s) (for Example **Capacitor;Resistor**) with a semi-colon separator between them followed by a pipe separator and enter the last update date in the following format:

YYYY-MON-DD HH24:MI:SS e.g. (**2011-JAN-01 00:00:00**)

This date parameter enables you to do incremental processing.

7. The input values let the DSA know which ones it should pull down from the Agile PLM system using the Agile EDQP Connector then click **OK**.

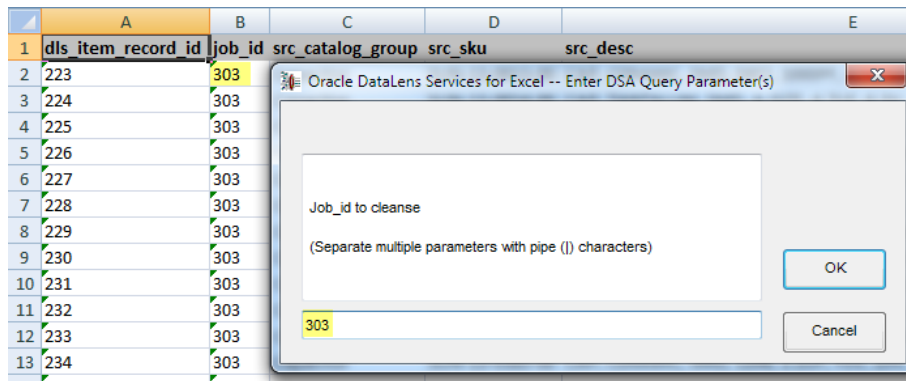


8. After the results are retrieved, click **OK** to return to the worksheet to see the results.

	A	B	C	D	E
1	dls_item_record_id	job_id	src_catalog_group	src_sku	src_desc
2	223	303	Capacitor	SUN-13-0013-00	CAP, CERAMIC, SMD, 0402, 1000PF, X7R, 50V, 10%
3	224	303	Capacitor	SUN-13-0014-00	CAP, TANTALUM, SMD, A-SIZE, 4.7UF, 6.3V, 20%
4	225	303	Capacitor	SUN-13-0016-00	CAP, CERAMIC, SMD, 0402, 20PF, COG, 50V, 5%
5	226	303	Capacitor	SUN-13-0033-00	CAP, CERAMIC, SMD, 0603, 0.01UF, Y5V, 50V, -20/+80%
6	227	303	Capacitor	SUN-13-0044-00	CAP, CERAMIC, SMD, 0402, 330PF, X7R, 50V, 10%
7	228	303	Capacitor	SUN-13-0045-00	CAP, CERAMIC, SMD, 0402, 10PF, COG, 50V, +/-0.5PF
8	229	303	Capacitor	SUN-13-0052-00	CAP, CERAMIC, SMD, 0402, 100PF, COG, 50V, 5%
9	230	303	Capacitor	SUN-13-0099-00	CAP, CERAMIC, SMD, 0603, 2.2UF, Y5V, 6.3V, +80%/-20%

#### 4.5.2 Cleanse Production Batch

7. Select the 20\_Batch\_Creation\_Details worksheet created in the previous step to begin the process of cleansing the production batch.
9. Select **Job Options...** from the DataLens Tools menu to select the Data Service Application (DSA) to process the job on the EDQP DataLens server.
10. Select **APLM\_CLEANSE\_PRODUCTION\_BATCH** from the DSA drop down and click **OK**.  
The APLM\_CLEANSE\_PRODUCTION\_BATCH DSA includes an example data lenses called APLM\_Capacitors, which is fully refined version.
11. Click the run button to initiate the job processing.
12. Enter the Job\_id (for Example **303**) from the results of the APLM\_CREATE\_PRODUCTION\_BATCH worksheet run then click **OK**.

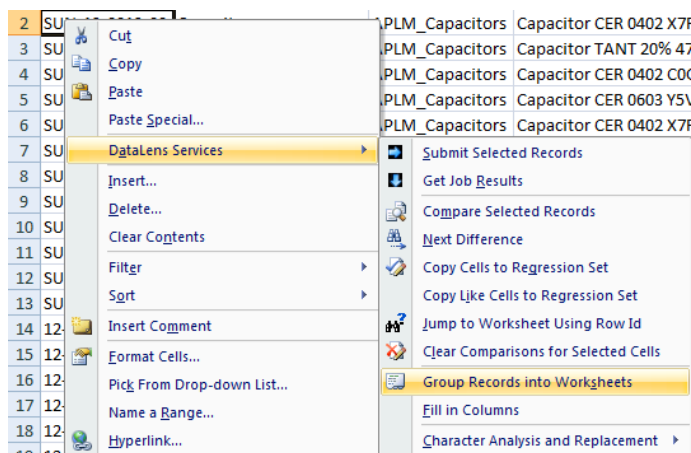


13. After the results are retrieved, click **OK** to return to the worksheet to see the results.

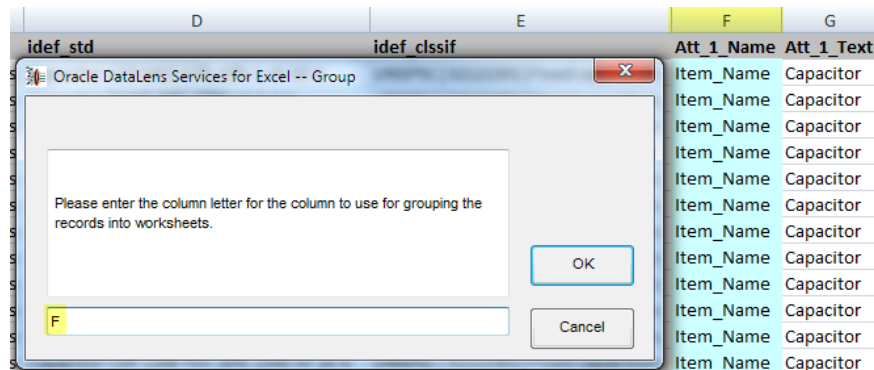
	A	B	C	D	E	F	G	H	I
1	src_sku	Item_Definition_Name	taxonomy	idef_std	idef_cisif	Att_1_Name	Att_1_Text	Att_2_Name	Att_2_Text
2	SUN-13-0013-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 X7R 10% 1 nF 50 V	UNSPSC 32121501 Fixed capacitors	Item_Name	Capacitor	Material	CER
3	SUN-13-0014-00	Capacitor	APLM_Capacitors	Capacitor TANT 20% 4700 nF 6.3 V	UNSPSC 32121501 Fixed capacitors	Item_Name	Capacitor	Material	TANT
4	SUN-13-0016-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 COG 5% 0.02 nF 50 V	UNSPSC 32121501 Fixed capacitors	Item_Name	Capacitor	Material	CER
5	SUN-13-0033-00	Capacitor	APLM_Capacitors	Capacitor CER 0603 Y5V 80% 10 nF 50 V	UNSPSC 32121501 Fixed capacitors	Item_Name	Capacitor	Material	CER
6	SUN-13-0044-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 X7R 10% 0.33 nF 50 V	UNSPSC 32121501 Fixed capacitors	Item_Name	Capacitor	Material	CER
7	SUN-13-0045-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 COG 0.01 nF 50 V	UNSPSC 32121501 Fixed capacitors	Item_Name	Capacitor	Material	CER

The attribute results are in name value pairs, which are not in the correct format for import into Agile.

14. Right-click on a cell in the spreadsheet, select **DataLens Services...** and then **Group Records into Worksheets** to normalize the attribute name / value column pairs.



15. Enter the column letter (. F) that represents the beginning of the attribute information.



16. Enter a filename and designate a location for the new spreadsheet then click **Save**. A message is displayed so that you know the spreadsheet has been created.
17. Click **OK**. The resulting normalized worksheet will have attribute headers and row values.

	A	B	C	D	E	F	G	H	I	J	K	L
	src_sku	Item_Definition_Name	taxonomy	idef_std	idef_clsif	Item_Name	Material	Package	Temperature_characteristics	Tolerance	Capacitance	Voltage
1	SUN-13-0013-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 X7R 10% 1 nF 50 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0402	X7R	10%	1 nF	50 V
2	SUN-13-0014-00	Capacitor	APLM_Capacitors	Capacitor TANT 20% 4700 nF 6.3 V	UNSPSC 32121501 Fixed capacitors	Capacitor	TANT			20%	4700 nF	6.3 V
3	SUN-13-0016-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 CG 5% 0.02 nF 50 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0402	CG	5%	0.02 nF	50 V
4	SUN-13-0033-00	Capacitor	APLM_Capacitors	Capacitor CER 0603 YSV 80% 10 nF 50 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0603	YSV	80%	10 nF	50 V
5	SUN-13-0044-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 X7R 10% 0.33 nF 50 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0402	X7R	10%	0.33 nF	50 V
6	SUN-13-0045-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 CG 0.01 nF 50 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0402	CG		0.01 nF	50 V
7	SUN-13-0052-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 CG 5% 0.1 nF 50 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0402	CG	5%	0.1 nF	50 V
8	SUN-13-0099-01	Capacitor	APLM_Capacitors	Capacitor CER 0603 YSV 80% 2200 nF 6.3 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0603	YSV	80%	2200 nF	6.3 V
9	SUN-13-0252-00	Capacitor	APLM_Capacitors	Capacitor CER 0603 YSV 80% 2200 nF 6.3 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0603	YSV	80%	2200 nF	6.3 V
10	SUN-13-0338-00	Capacitor	APLM_Capacitors	Capacitor CER 0402 YSV 20% 220 nF 10 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	0402	YSV	20%	220 nF	10 V
11	SUN-13-0385-00	Capacitor	APLM_Capacitors	Capacitor CER 1206 YSV 20% 2200 nF 25 V	UNSPSC 32121501 Fixed capacitors	Capacitor	CER	1206	YSV	20%	2200 nF	25 V

## 4.6 Importing Cleansed Data back into Agile

Once the Excel worksheet is created, the standard Agile PLM Import process should be used to load the results back into Agile PLM. You must develop an overall data cleansing processes that takes into account their unique requirements.

### 4.6.1 Basic Import Process

Following is an example of the process to import the content into Agile PLM without any additional ECO process.

1. Start the Import Wizard and load the Excel file.
2. Select the right worksheet.
3. Import the Item data.
4. Select the import without ECO and Map option on the next screen.
5. Map the attributes to be imported.
6. Complete the import.

This sequence is illustrated in the following images:

Import - Mozilla Firefox

Go to a Website

## Import

**File Selection** 1 2 3 4

Select a file to import. The following file types are supported: Excel (xls), Text (txt / csv), Product Data Exchange (pdx) and Agile XML (axml). [Help Link](#)

\* Import File: D:\MyDocuments\0\_data\0\_ProdMgmt\9.3.2\PDQ\Ag [Browse...](#)

Excel File Configuration Options

Template Type: Standard, no template

Select Worksheet: Item\_Name

Location of Header Row: 1

Location of Last Row: N/A

Preferences < Back Next > Import Validate Cancel

Import - Mozilla Firefox

sci34005.us.oracle.com:7001/Agile/ImportServer

## Import

**Specify File Content** 1 2 3 4 5

Specify the content to import from the options below. [Help Link](#)

- ▶ Manufacturer
- ▶ Manufacturer Part
- ▼ Item
  - ☒ Item only
  - ☐ Approved Manufacturer List
  - ☐ Sites
  - ☐ Relationships
  - ☐ Bill of Material
  - ☐ Specification
  - ☐ Approved Supplier List
  - ☐ Instances
- ▶ Customer
- ▶ Published Price
- ▶ Quote History
- ▶ Part Group
- ▶ User Group
- ▶ User
- ▶ PSR
- ▶ Quality Change Requests
- ▶ Supplier
- ▶ Declaration
- ▶ Substance
- ▶ Specification
- ▶ Project
- ▶ Currency Conversion

Preferences < Back Next > Import Validate Cancel

Import - Mozilla Firefox

sc34005.us.oracle.com:7001/Agile/ImportServer

## Import

Select Content Options 1 2 3 4 5

Select a mapping and data transformation method. If you are importing Item or Price content specify whether or not a Change Order is required for modifications. [Help Link](#)

▼ Data Mapping

☒ Define attribute mapping in next step

☐ Use a saved mapping file

[Browse...](#)

▼ Data Transformation

☒ No transformations are necessary

☐ Use a saved transformation file

[Browse...](#)

[Download transformation template](#)

▼ Redline Options

☒ Updates do not require redlining

☐ Use the following Change for redlining

[Search](#) [Add](#)

[Preferences](#) [Back](#) [Next >](#) [Import](#) [Validate](#) [Cancel](#)

Import - Mozilla Firefox

sc34005.us.oracle.com:7001/Agile/ImportServer

## Import

Specify Attribute Mapping 1 2 3 4 5

Specify and review attribute mapping definition. [Help Link](#)

**Import Fields**

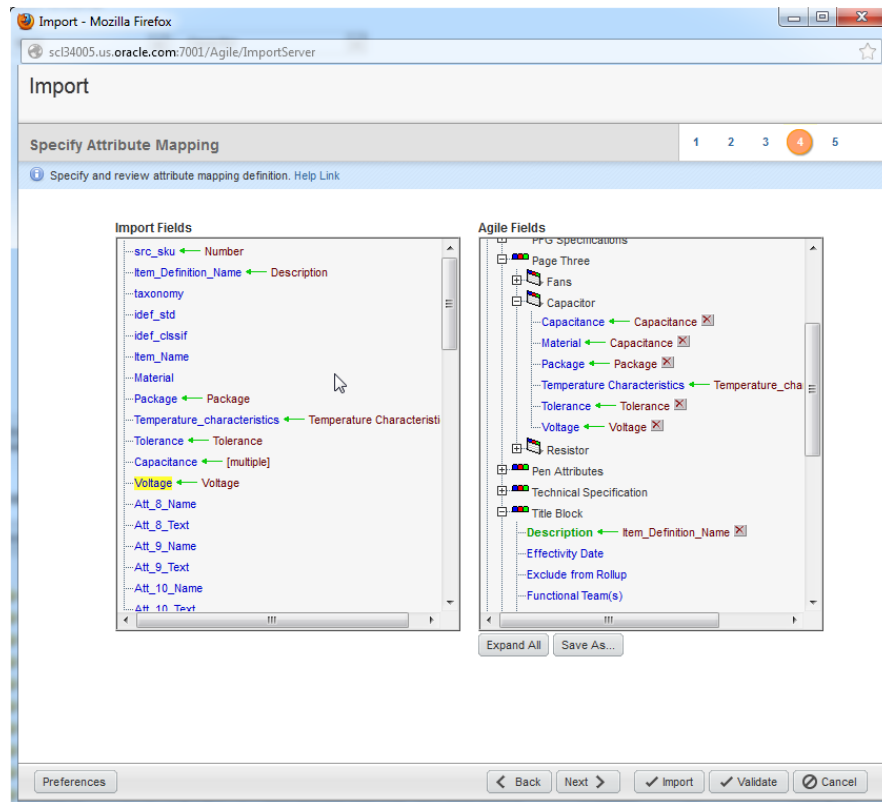
- src\_sku ← Number
- Item\_Definition\_Name ← Description
- taxonomy
- idef\_std
- idef\_clsif
- Item\_Name
- Material
- Package
- Temperature\_characteristics
- Tolerance
- Capacitance
- Voltage
- Att\_8\_Name
- Att\_8\_Text
- Att\_9\_Name
- Att\_9\_Text
- Att\_10\_Name
- Att\_10\_Text
- Att\_11\_Name

**Agile Fields**

- Technical Specification
- Title Block
  - Description ← Item\_Definition\_Name
  - Effectivity Date
  - Exclude from Rollup
  - Functional Team(s)
  - Item Group(s)
  - Lifecycle Phase
  - Mass
  - Mass-measure
  - Number ← src\_sku
  - Part Category
  - Part Family
  - Part Type
  - Project
  - Rev
  - Shippable Item
  - Sites
  - Size

[Expand All](#) [Save As...](#)

[Preferences](#) [Back](#) [Next >](#) [Import](#) [Validate](#) [Cancel](#)



## 4.6.2 Additional process considerations

When designing the overall cleansing process, user privileges and attribute configuration will determine who can perform the import and when it can be done. For example, if some of the data is in Change Controlled Attributes, then a Change Order or Manufacturer Order will be required to import the data and then complete the changes. Only a user who has modify rights to all the data will be able to successfully complete an import.

A company may also want to prenotify key constituent groups before completing an import that changes a large amount of data. With this type of communication, users will not be surprised to see new or altered attributes. If you are updating descriptions across 3,000 capacitors, all users will need to be informed so that they can perform searches and other tasks without becoming frustrated with the data changes.

Another issue is synchronization. When changes are made in Agile PLM, they also need to be communicated to other enterprise systems, like ERP. How this communication occurs will be influenced by the number of changes being made and the nature of the changes. If Agile only data is changed, like Tolerance on Page 3, then there will not need to be any communication. However, if 50,000 descriptions are updated, then the communication method of that scope of change should be considered. Can the direct ERP integration process a change of that many descriptions? Should ERP also be updated at the same time as Agile?

As with any process that impacts your production data, it is always wise to test and validate the process in a Development or QA system before proceeding to a large production import. Once

the overall process is stable and well understood, then proceed to resolve data issues in Agile PLM as an ongoing practice.

In summary, spend time planning the overall data change process, including technical details, communication, integration, and testing, as you implement this new capability to leverage EDQP with Agile PLM.

*Copyright 2013, Oracle and/or its affiliates. All Rights Reserved.*

This document is provided for information purposes only, and the contents hereof are subjected to change without notice. This document is not warranted to be error-free, nor is it subjected to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document; and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose without prior written permission from Oracle.