

Oracle Enterprise Data Quality

Hands-on-Lab 7653

Oracle Openworld 2017



Table of Contents

Scenario.....	3
Part 1 – Launch the Director User Interface.....	4
Part 2 – Profiling the data using EDQ Product Data Services	6
Part 3 – Matching the data using EDQ Product Data Services	12
Part 4 – Product Classification	17
Part 5 – Extracting Attributes from the Description for a Category of Products.....	21

Scenario

Our company takes incoming data about the products it sells, and stores it in a central product data hub. The most important field in the incoming data is a free-text description that contains key information about the products. There is no standard format for this free-text field. It usually contains similar sorts of information – for example, units of measure, types of product, colors, dimensions, materials, the product's name, and so on. But this information is often jumbled up in different orders, and a variety of abbreviations often used to refer to the same properties (for example, STEEL and STL).

Here are two examples of typical values:

1 1/8-12X4 HX HD CAP SCR-GR 8 ZINC PL(LE)

1/2-13X4 1/2 HX HD FULL THRD CAP SCR GR 2

The product hub requires structured data records. Therefore attributes of each product must be extracted from the incoming data's description field and standardized. In addition, any duplicate records must be identified and merged.

The challenge is to understand the data, match it, and restructure it as follows:

Original Description	Base Description	Category	Head Type	Material	Screw Type	Dimensions	Grade
1 1/8-12X4 HX HD CAP SCR-GR 8 ZINC PL(LE)	HEX HEAD CAP SCREW-ZINC PL(LE)	SCREW	HEX HEAD	ZINC PLATED	CAP	1 1/8-12X4	GR 8
1/2-13X4 1/2 HX HD FULL THRD CAP SCR GR 2	HEX HEAD FULL THRD CAP SCREW	SCREW	HEX HEAD		CAP	1/2-13X4 1/2	GR 2


In order to achieve this, EDQ's main tasks are:

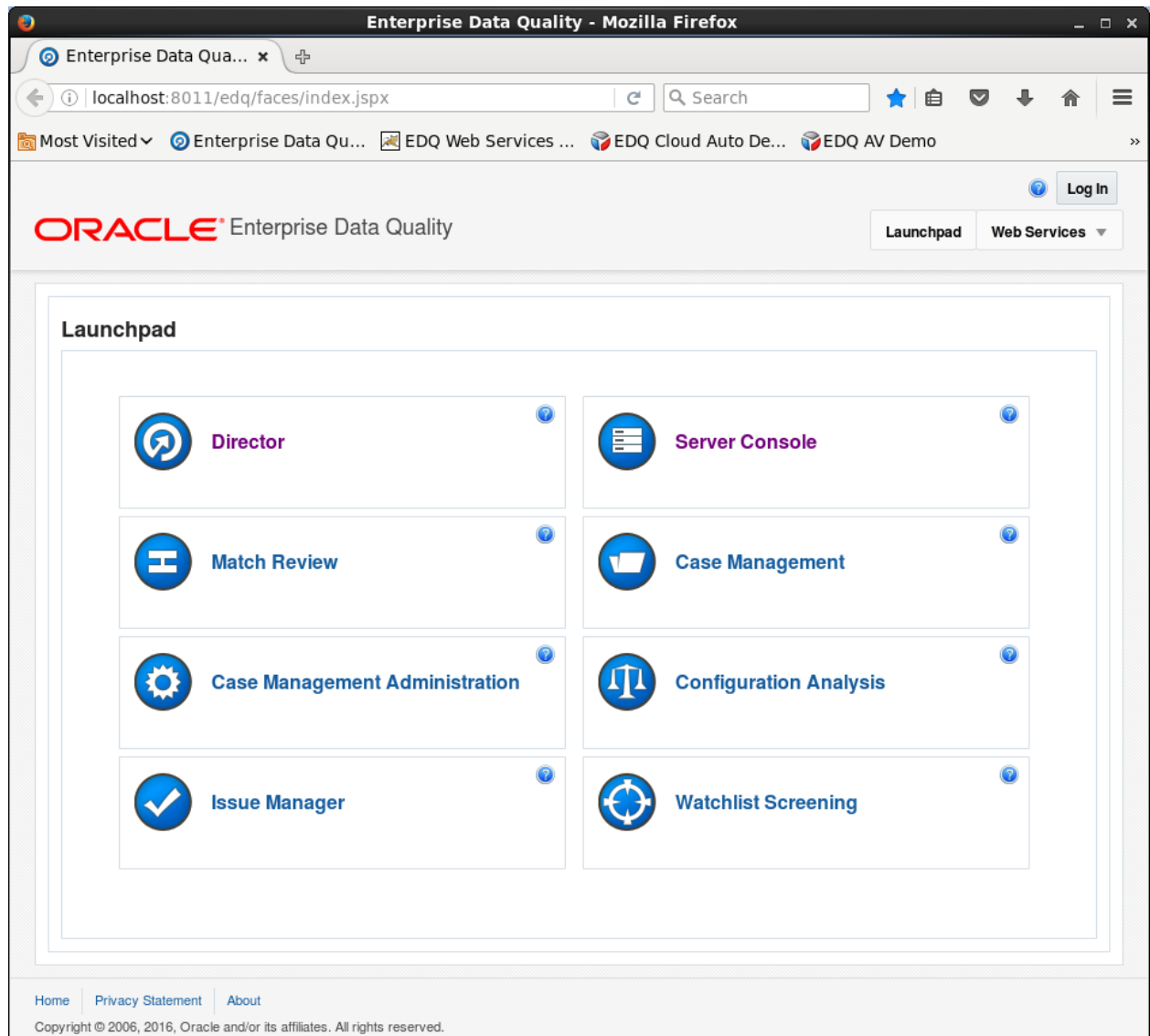
- Profiling the description field to understand its contents
- Classifying each record as belonging to a particular category of product (SCREW, BOLT, MOTOR and so on)
- Recognizing key pieces of information within the description field
- Extracting these attributes into their own fields
- Standardizing the extracted data
- De-duplicating the standardized data records

Part 1 – Launch the Director User Interface

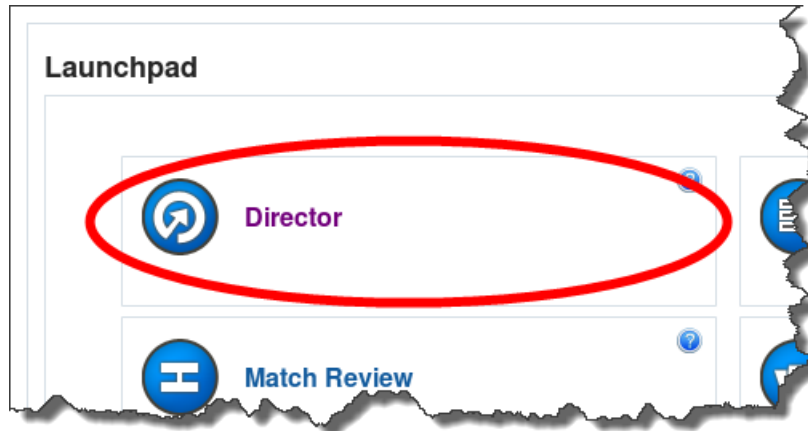
A - Launch the Director UI



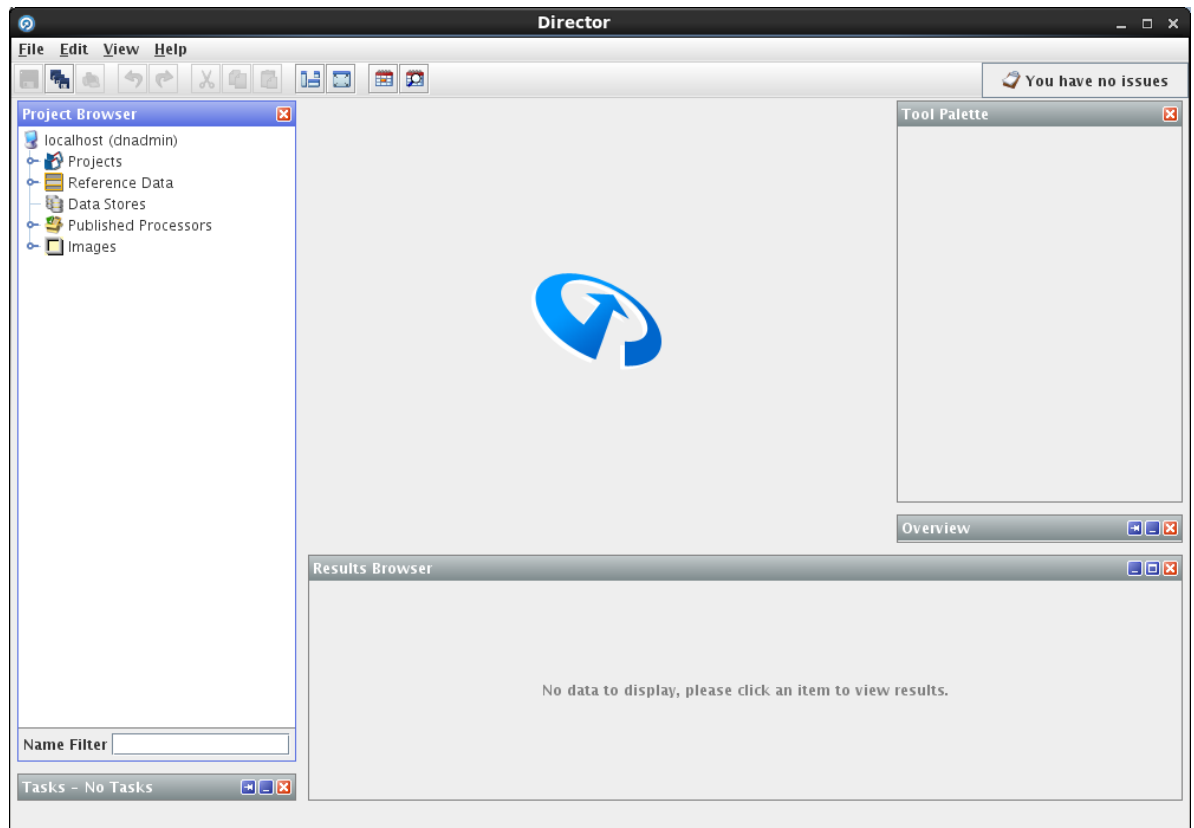
1. Double click the Enterprise Data Quality icon on the desktop () or open the Firefox Web Browser and navigate to the following URL: <http://localhost:8011/edq>. The Enterprise Data Quality Launchpad is displayed:



2. Click the Director link to launch the Director user interface.



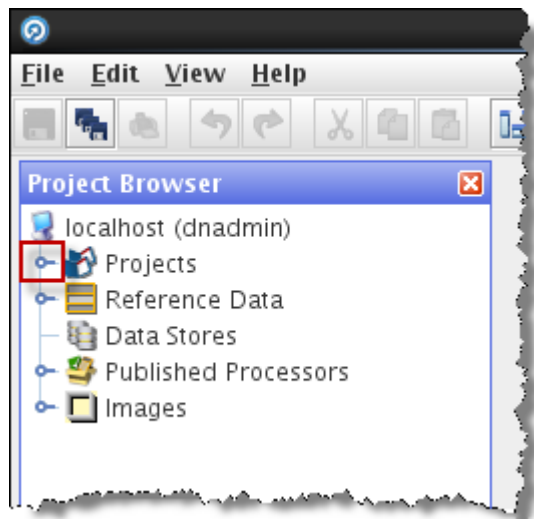
3. If a dialog box tells you that a Java update is needed, click **Later**.
4. A dialog box will ask you whether you want to run this application, click the 'I accept the risk...' check box, and click **Run**.
5. Login with the username **dnadmin** and password **dnadmin**.



Part 2 – Profiling the data using EDQ Product Data Services

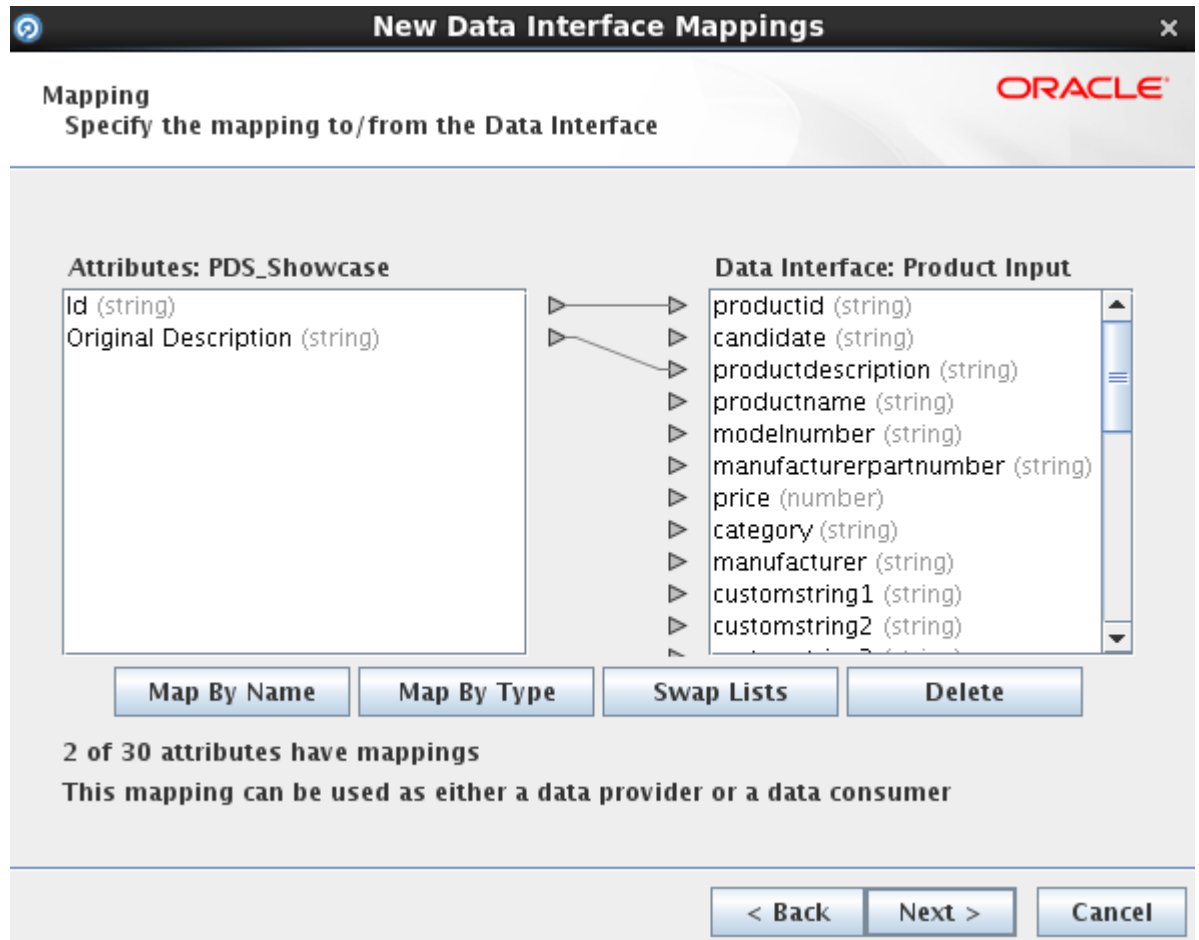
A – Map the data to the Product Input data interface and run the provided profiling process.

1. In the Project Browser, on the left-side of the Director UI, expand the Projects node.

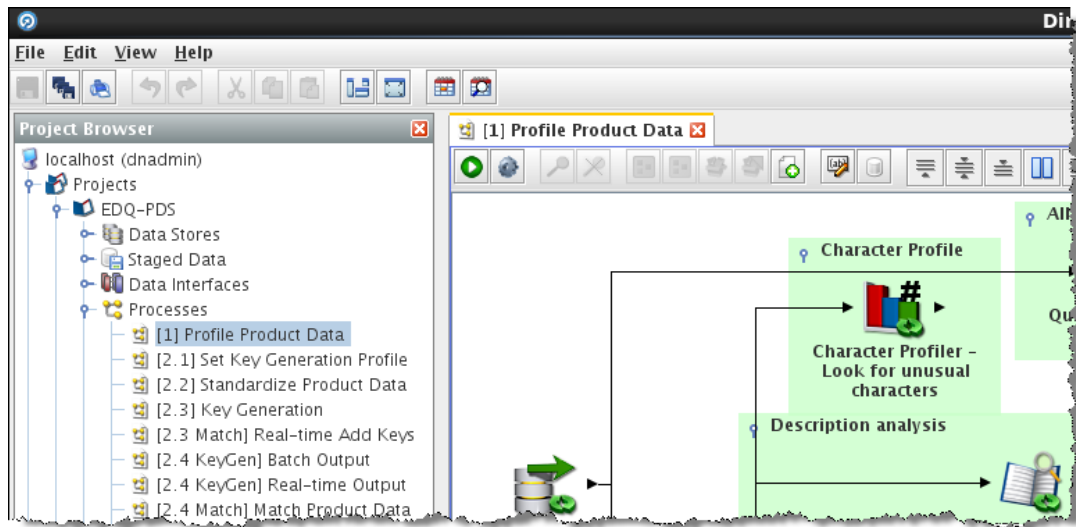


2. Expand the **EDQ-PDS** project, and, within that project, expand the **Reference Data** node.

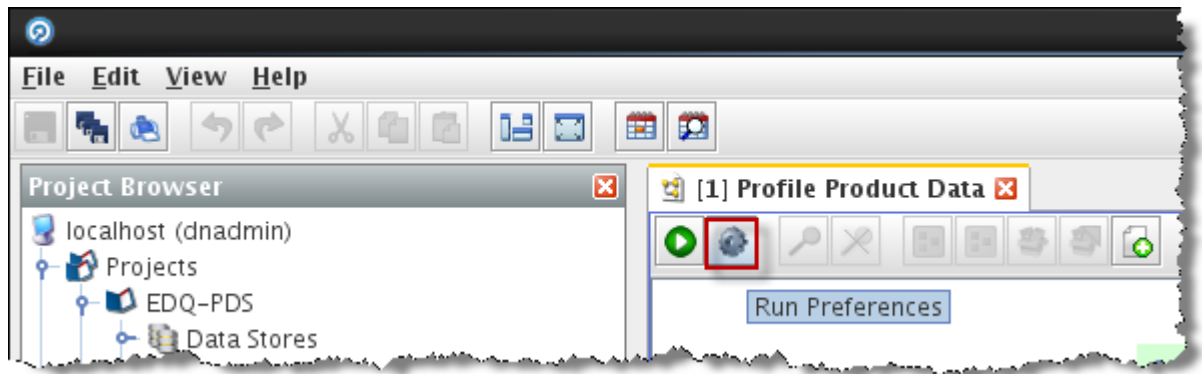
Note: In this case, we have captured some sample data as Reference Data in the EDQ project. Normally data is captured from source using a Snapshot and would appear under the Staged Data node in the Project tree.
3. Right-click on the **PDS_Showcase** reference data set (our sample data set) and select **Create Data Interface Mapping...**
4. Select the **Product Input** data interface. This interface is used by the out-of-the-box profiling and matching processes provided in the EDQ Product Data Services Pack. Click **Next**.
5. Click on the **Swap Lists** button to show the data attributes on the left and the Data Interface on the right (as we are mapping data 'IN' in this case).
6. Map the **Id** field from the sample data to the **productid** field on the interface, and the **Original Description** field from the sample data to the **productdescription** field as shown below:



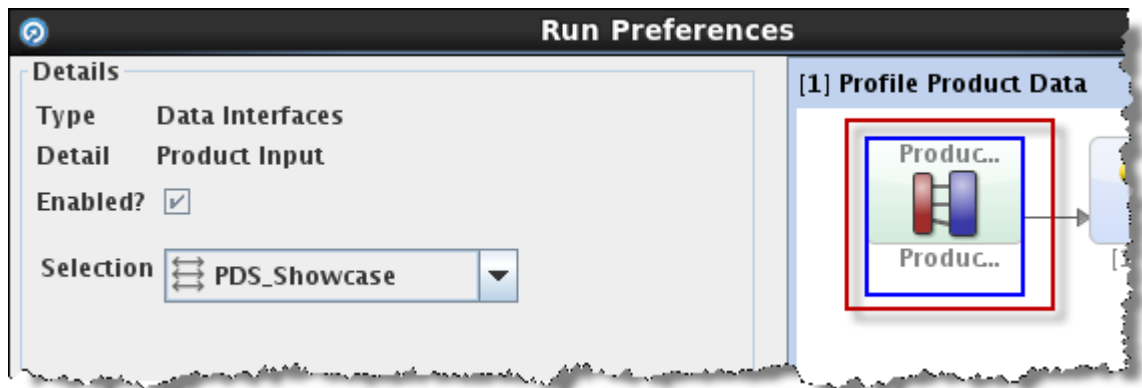
- Click **Next**. Keep the default name for the data interface mapping (PDS_Showcase), and click **Finish**.
- Now expand the **Processes** node and double-click on the **[1] Profile Product Data** process to open it in the Canvas (the middle area of the screen):



- We need to change this process to read from our sample data using the new Data Interface mapping. To do this, use the cog button to bring up the Run Preferences for the process:



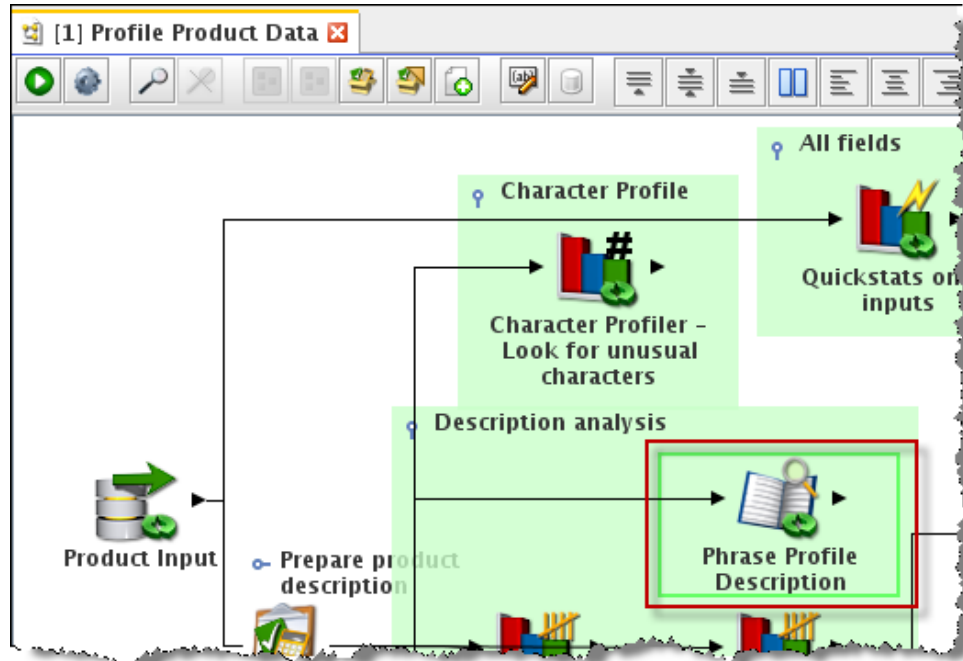
- Click on the Data Interface on the right-hand side of the screen, then change the mapping Selection to PDS_Showcase on the left-hand side, as shown below:



- Now, click on the Run button to save this change and run the process from our sample data.

B - Examine the Profiling Results

- In the Canvas, click on the **Phrase Profile Description** processor:



- The area of the screen directly below the Canvas is called the Results Browser. In the Results Browser you can now see the results of this phrase profiler, showing all the common words and phrases in our input product descriptions.

Size	Top Phrase	Phrase	Frequency
1		X	66
1		SCREW	58
1		CAP	47
1		HD	41
1		HEAD	34
1		SS	33
1		SET	31
1		ZINC	30
1		SCR	29
1		MOTOR	26
1		BOLT	25
1		ALLOY	24
1		FLAT	24
1		HEX	24
2		18-8 SS	23
1		SOCKET	23

3. The profiling process also looks for specific things within the input product description, such as English words, non-English words, possible abbreviations, tokens with numbers etc. We can use these views to build up reference data in order to classify items and extract product attributes, and use the drilldowns to ensure we classify tokens correctly, avoiding ambiguities.
4. Now click on the processor labelled **Quickstats on all inputs** and examine its results. Note the following:
 - a. There are a total of 233 records in our sample product data.
 - b. The productid is fully populated and always unique.
 - c. The productdescription is also fully populated and has 12 duplicates
5. Have a tour round a few of the profiling views, such as **Possible Abbreviations**, and **English Words**.
6. Note that the profiling process also includes the use of provided reference data sets to perform some initial extraction of potential attributes from the data. For example, scroll to the right hand side of the screen and click on the **Extracted Units of Measure** profiler to look at the different units of measure and their quantities in the data. This can help us determine how much work we might need to do to standardize and match the data.

C - What have we learned from Profiling?

- The data contains rows relating to several different types of product:

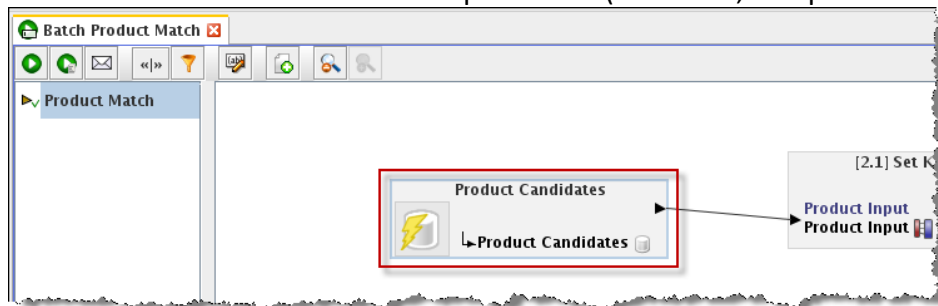
- Screws
- Bolts
- Capacitors
- Motors
- Resistors
- The description field contains many quantified units of measure, in different patterns.
- The data contains different abbreviations of the same value (**screw** and **scr**). Our process will need to first recognize these as meaning the same thing and then standardize them.

Part 3 – Matching the data using EDQ Product Data Services

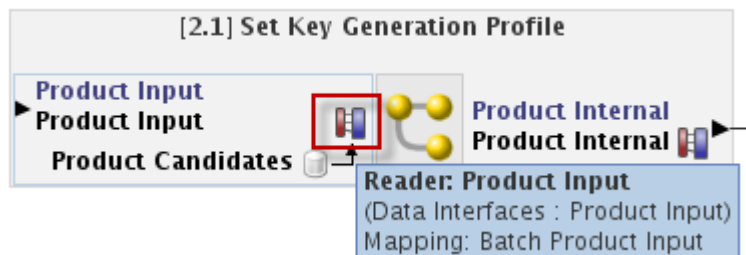
In this part, we will map our data into the out-of-the-box product data matching service provided in EDQ 12.2.1.3 and review results. In this way we can gain insight into how many actual duplicates we might have in the data using more advanced techniques than in profiling (where there were 12 records with exactly the same product description).

A – Change the configuration of the batch matching job to read in our mapped data

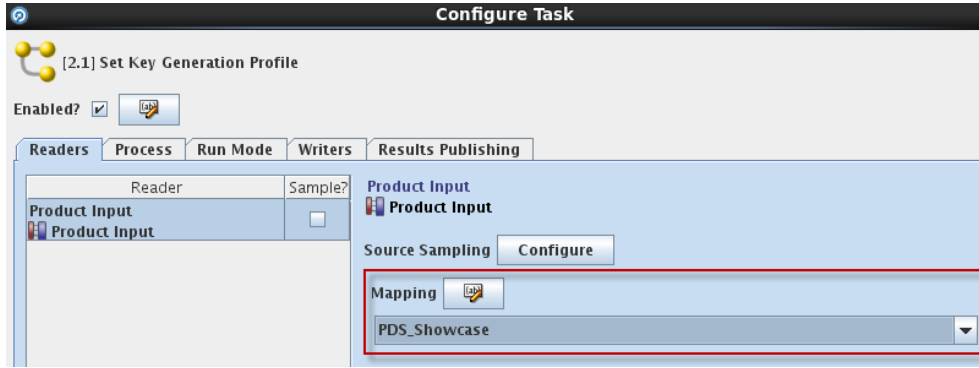
1. In the EDQ-PDS project, expand the Jobs node and double-click on the job named **Batch Product Match**
2. By default, this takes its input data from a table in the EDQ Staging Database. This is for use when the product data services are integrated with external applications. In this case we have the data in EDQ, so we can change this using the following steps:
 - a. Delete the **Product Candidates** snapshot task (click on it, and press the Delete key):



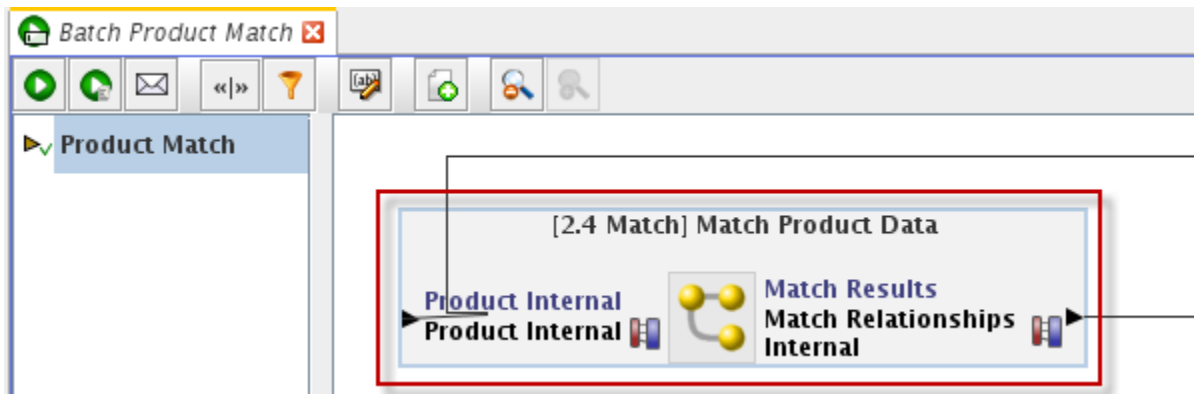
- b. Double-click on the **Product Input** data interface of the **[2.1] Set Key Generation Profile** task:



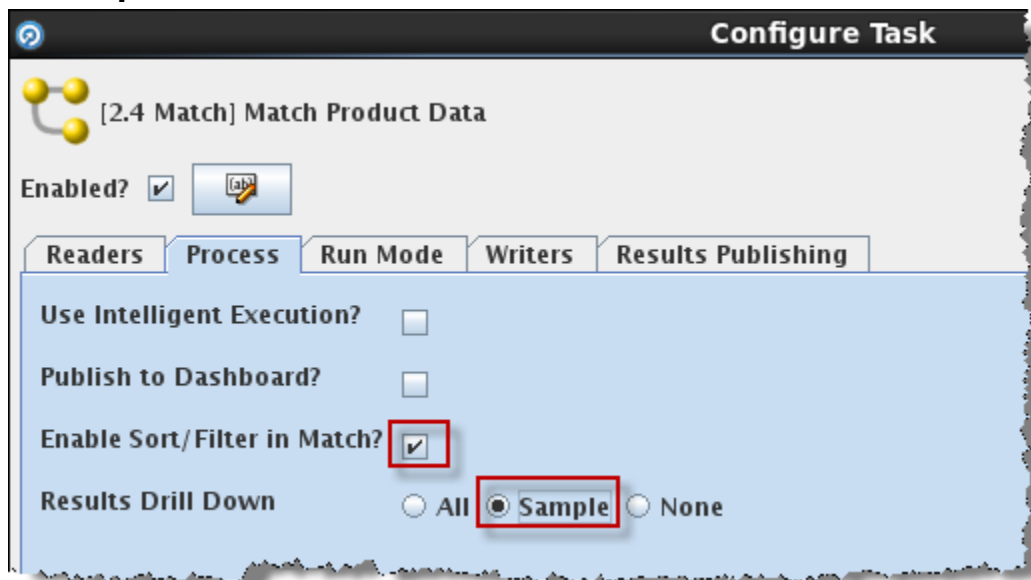
- c. In the Mapping section on the right-hand side, select **PDS_Showcase** as the data interface mapping to use. This will map in the Id and Original Description fields from our sample data as previously:



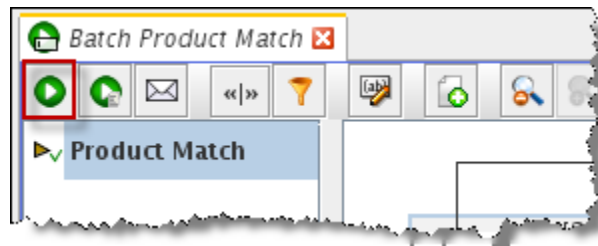
3. We also need to change the job to make sure it generates output that we can review in the Match Review screen. To do this, scroll down in the job definition until you find the task named **[2.4 Match] Match Product Data** and double-click on it to configure it.



4. Tick the option to **Enable Sort/Filter in Match?** and change the **Results Drilldown** option to **Sample** rather than **None** as shown below:



5. Click the OK button and then click on the 'play' button to run the job



6. The job will then take a few minutes to run. You can see the progress in the Tasks window and in the centre of the screen. When it is complete, right-click on the **[2.4 Match] Match Product Data** task and click on **Open...** to open the process and review its results.

7. Click on the **Product Data Match** processor. The Rules tab is shown, showing 14 strong matches, and 260 intermediate matches:

Product Data Match

- ▶ Groups
- ▶ Relationships
- ▶ Merged
- ▶ Decisions

Process

Results Browser

Job: **Batch Product Match**

Rule Order	Match Rule	Relationships
1	UID1001 UID1 exact	0
2	UID2001 UID2 exact	0
3	UID3001 UID3 exact	0
4	EID1 no match	0
5	EID2 no match	0
6	EID3 no match	0
7	IEID1 Match	0
8	IEID2 Match	0
9	IEID3 Match	0
10	Overall score (strong match)	14
11	Overall score (intermediate match)	260
12	Overall score (weak match)	0

8. To review these results and see the matches, we can use EDQ's Match Review screen. To do this, double-click on the Product Data Match processor to open it, then click on the **Review Results** link:

- Advanced Options

Review Results

Configure Bulk Review Rules

Delete Realtime Review Results
- Assign Relationship Review

Assign Merged Review

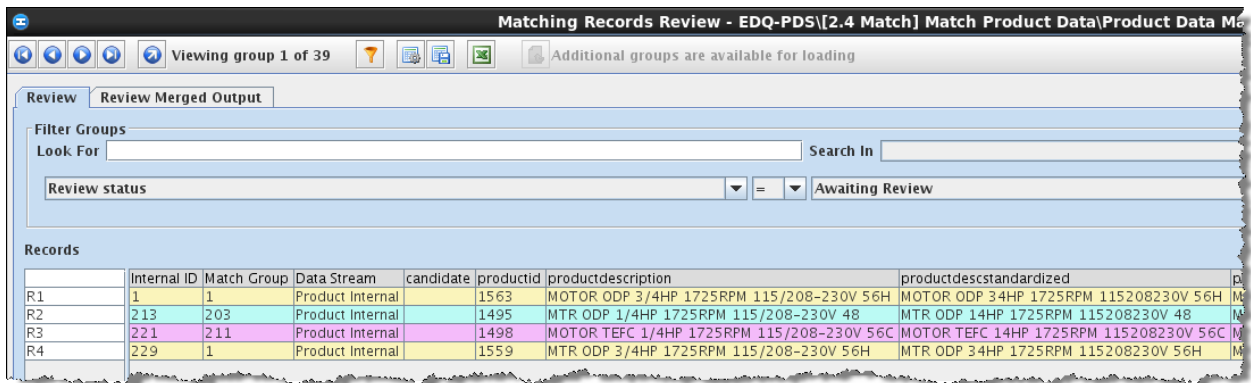
View Match Statistics

Delete Manual Decisions

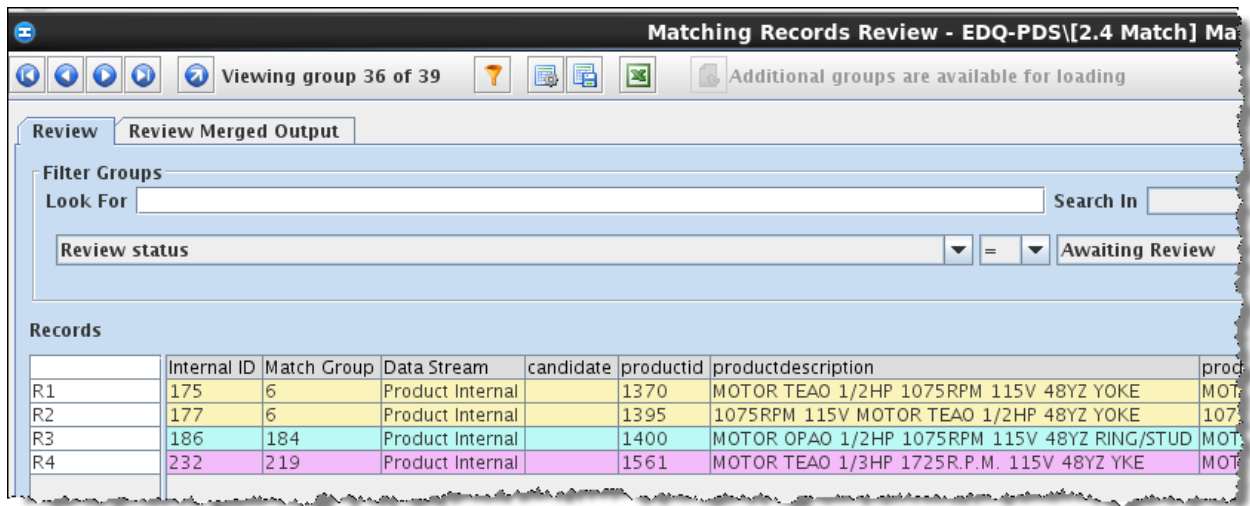
Externalize

9. This will open a separate Match Review screen. Maximize the window to see the results fully.

10. Note the first group that appears. A strong match has been found between two motors that share the same units of measure. These two records have the same background (yellow) as they have been automatically grouped together. The other matches are only candidate matches – a user would have to decide whether they were true matches or not. This is because although the descriptions are similar there are significant unit differences in the data (in this case different HP ratings and trailing numbers):



11. Scroll through the groups using the arrows in the top left corner of the screen to show some more match results. Notice that although at this stage we have not extracted and standardized the data using any of our own logic, EDQ is able to match product descriptions intelligently, for example distinguishing between strong matches where product dimensions are the same but the details of the product appears in a different order, and weaker matches where the dimensions or model are different:



12. We can now determine how hard we need to work (or otherwise) in restructuring the data if our main goal is to match it. Note that in many cases, EDQ's out-of-the-box product data matching service can be used directly to provide effective batch and real-time match results with no need for additional tuning.

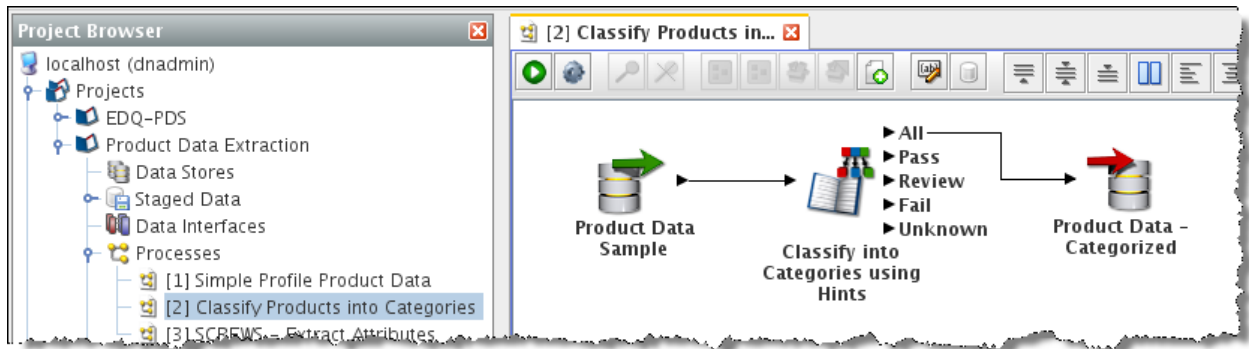
13. However, in this case, we know we need to restructure the data as well, so close the Match Review application and in the next part we will show how to use EDQ to do this.


Part 4 – Product Classification

In this part, we can look at a worked example of how it is often helpful to classify products into categories of item, and then work on parsing, extracting and standardizing data on a per category basis, since any given category of products will normally have a certain set of attributes that we need to extract.

A - Examine the Classify Products Process

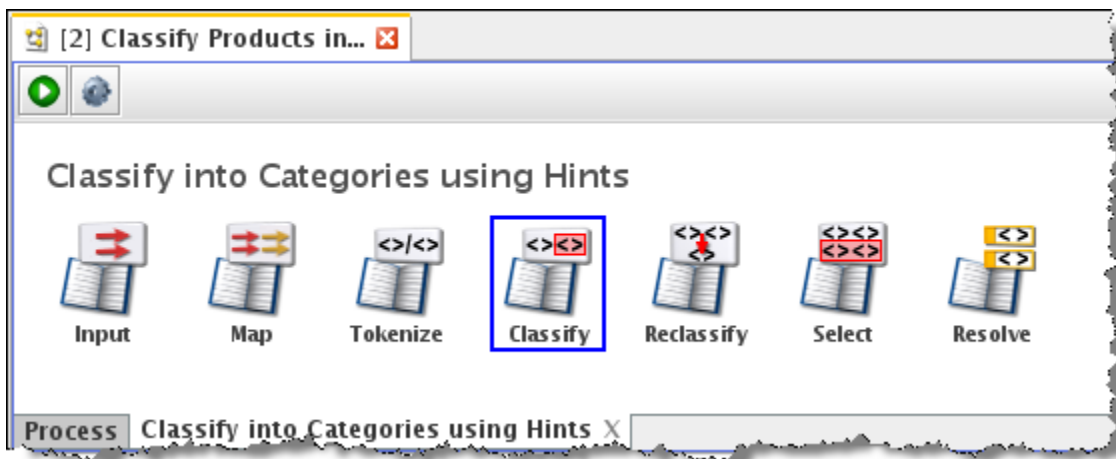
1. In the Project Browser, on the left-side of the Director UI, expand the **Product Data Extraction** project and double click the **[2] Classify Products into Categories** process to open it on the Canvas:



2. Near the top-left of the Canvas, click  to run the process.
3. In the Canvas, click the first (Reader) processor, labeled Product Data Sample. In the Results Browser you can now see the Id and Original Description fields. This is the same data we have worked with in the previous parts of the lab.
4. In the Canvas, click the processor labeled **Product Data - Categorized** (a Writer processor).
5. Look at the Results Browser, and note that the data that is written out of this process includes an extra column: **Category**.
6. Use the scroll bar on the right-side of the Results Browser to examine the various categories that have been identified.

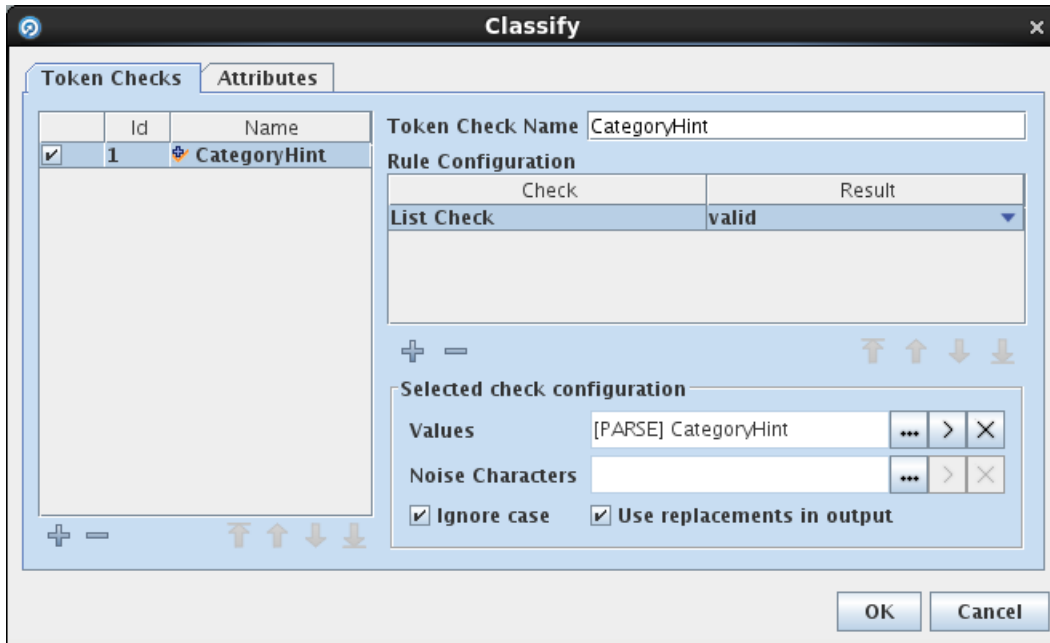
Id	Original Description	Category
227	RES 0603 1% 169 OHM 3W	RESISTOR
1563	MOTOR ODP 3/4HP 1725RPM 115/208-230V 56H	MOTOR
232	RES 0603 3.5W 1% 169 OHM	RESISTOR
1526	1/4-20 x 3/8 RH SI Machine Screw 18-8 SS	SCREW
233	RES 0805 0 OHM 1/4 W 2%	RESISTOR
1588	1/4-20 x 5/8 Cup Pt Socket Set Scr 316 SS	SCREW
235	resistor 2% 00hm 1/4w 0805	RESISTOR
1589	1/4-20x 1/2" FH SL MACHINE SCR NAT NYLON	SCREW
239	RESS , 1206 TF CH , 0.0 OHM , 1 / 8 W , 5%	RESISTOR
241	RESS 825 OHM 1/8 W 1%	RESISTOR
1590	1/4-20X1 3/4 Hx Hd Cap Screw Aluminum	SCREW
243	RESS 825 OHM 1/8 W 1%	RESISTOR
1592	1/4-20X1 ALLY 25/PK CONE PT SET SCREW	SCREW
1611	1/4-20X1/2 ALLOY 50/PK KNURL PT SET SCREW	SCREW


- You should find that most records have been assigned to one of the following categories: Screw, Bolt, Motor, Capacitor, Resistor.
- How have these categories been assigned? On the Canvas, double-click the processor labeled **Classify into Categories using Hints** (a Parse processor) to open it.
- Double-click the Classify sub-processor.

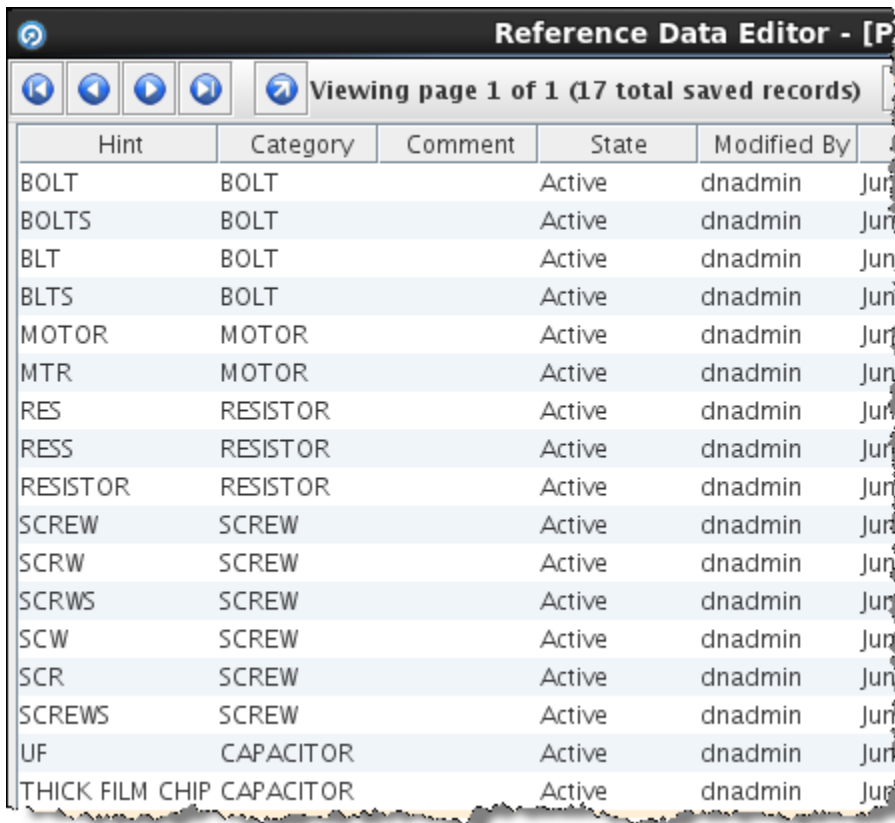


The Classify dialog opens.

10. Click on the **CategoryHint** Token Check.



11. Click  besides the Values field. The Reference Data Editor opens:



12. This parsing processor uses 'hints' to assign categories to rows of data. Both the hints and their associated categories are held in EDQ reference data. You can see them in the first

and second left columns above. If, for example, the processor finds 'SCREW', 'SCRW', 'SCRWS', 'SCW', 'SCR' or 'SCREWS' in the description field, then it assigns the data row to the 'SCREW' category. EDQ Product Data Services ships with sets of pre-prepared reference data, but you should also note that EDQ reference data is extensible: you can add new rows to existing reference data, and you can also develop whole new sets of reference data yourself if you need to.


13. Click Cancel to close the Reference Data Editor.
14. Click Cancel again to close the Classify dialog.

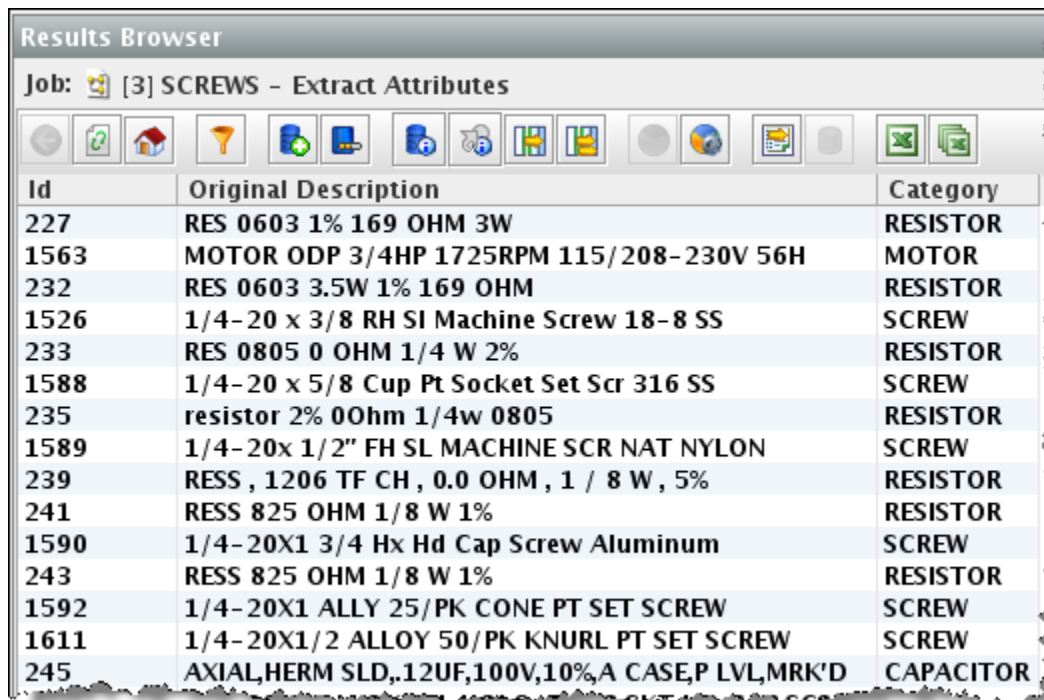
Let's summarize the [2] Classify Products process then: it assigns the data records to product categories by looking for category hints in the description field. Its dictionary of category hints is held in EDQ reference data. This reference data can be created or extended using the results of profiling.

Part 5 – Extracting Attributes from the Description for a Category of Products

In this final part, we will work on one of the main categories of product that we have in our data, and show how EDQ can apply different techniques to extract and standardize attributes from a product description. Finally, we will see how EDQ can write out different formats of product data depending on the needs of the downstream system.

A - Examine the Extract Attributes Process

1. In the Project Browser, on the left-side of the Director UI, double click the **[3] SCREWS – Extract Attributes** process to open it on the Canvas.
2. Near the top-left of the Canvas, click  to run the process.
3. In the Canvas, click on the processor labeled **Product Data - Categorized** (a Reader processor). In the Results Browser you can now see the Id, Original Description and Category fields. This process reads in the data set that the previous process wrote out.

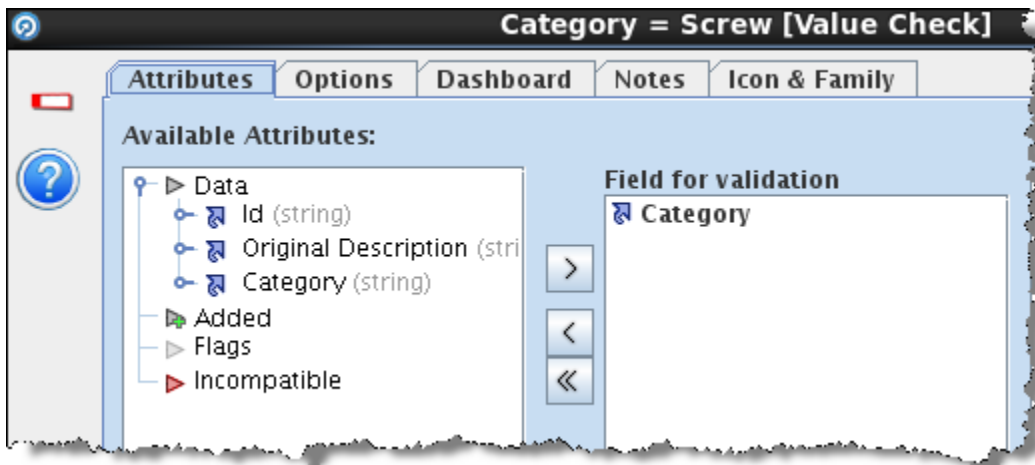


The screenshot shows the Results Browser interface with the job name "[3] SCREWS - Extract Attributes". The table below displays the extracted data with columns for Id, Original Description, and Category.

Id	Original Description	Category
227	RES 0603 1% 169 OHM 3W	RESISTOR
1563	MOTOR ODP 3/4HP 1725RPM 115/208-230V 56H	MOTOR
232	RES 0603 3.5W 1% 169 OHM	RESISTOR
1526	1/4-20 x 3/8 RH SI Machine Screw 18-8 SS	SCREW
233	RES 0805 0 OHM 1/4 W 2%	RESISTOR
1588	1/4-20 x 5/8 Cup Pt Socket Set Scr 316 SS	SCREW
235	resistor 2% 00hm 1/4w 0805	RESISTOR
1589	1/4-20x 1/2" FH SL MACHINE SCR NAT NYLON	SCREW
239	RESS , 1206 TF CH , 0.0 OHM , 1 / 8 W , 5%	RESISTOR
241	RESS 825 OHM 1/8 W 1%	RESISTOR
1590	1/4-20X1 3/4 Hx Hd Cap Screw Aluminum	SCREW
243	RESS 825 OHM 1/8 W 1%	RESISTOR
1592	1/4-20X1 ALLY 25/PK CONE PT SET SCREW	SCREW
1611	1/4-20X1/2 ALLOY 50/PK KNURL PT SET SCREW	SCREW
245	AXIAL,HERM SLD,.12UF,100V,10%,A CASE,P LVL,MRK'D	CAPACITOR

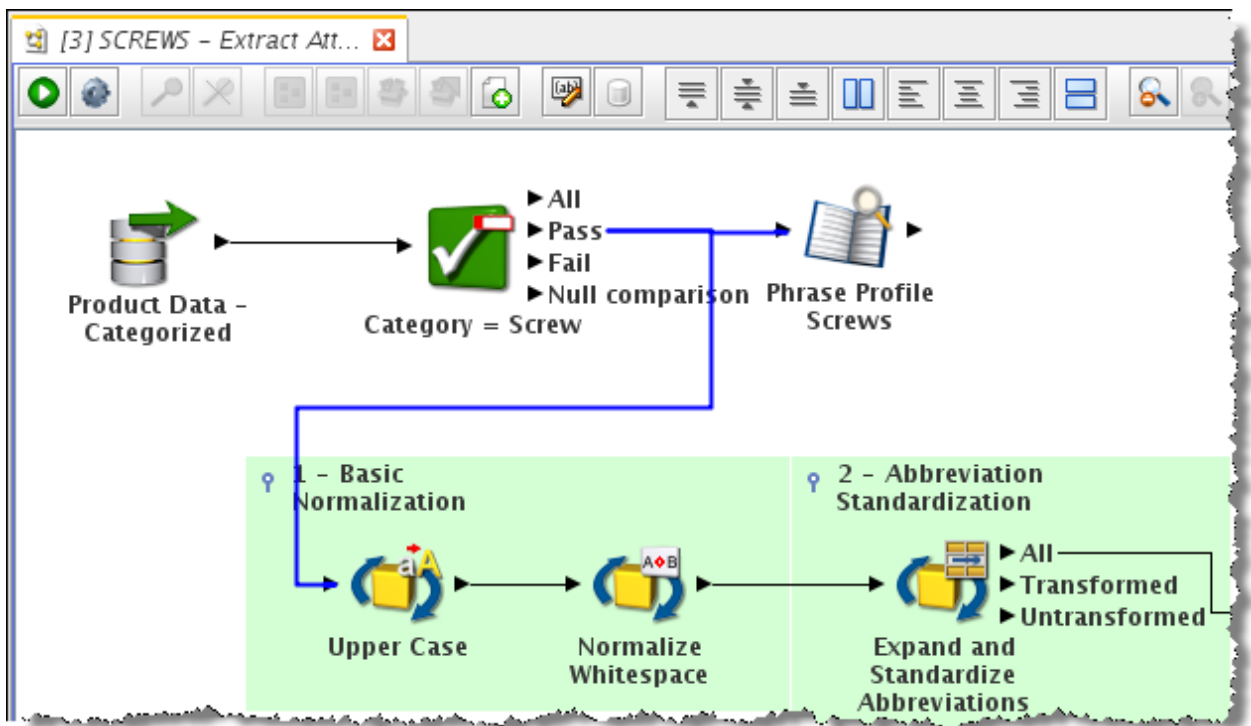
4. In the Canvas, double-click the processor labeled **Category = Screw** (a Value Check processor) to open it.

- Note that **Category** is the input field.



- Navigate to the **Options** tab.
- Note that the Value to compare records against is set to **SCREW**.
- Click **Cancel** to close the processor.

Note that only records that pass the value check in the Category = Screw processor are sent to downstream processors:



So, the process checks to see whether the Category is Screw. Effectively, the Value Check is a filter that only allows records with a category of SCREW to pass. Records for all other categories are not processed any further by this process.

- In the Canvas, click the **Upper Case** processor.
- Look at the Results Browser, and note that this processor standardizes all letters to UPPER CASE.

- In the Canvas, click the **Normalize Whitespace** processor. Note that this processor removes leading and trailing whitespaces, and also normalizes multiple consecutive concurrencies to a single whitespace.

So far, the process simply reads in the data, filters out all records that are not screws, and carries out some basic data normalization on the remaining records. The next stage is to extract attributes of screws from the product description field.

B – Understand different ways to Extract Attributes

This process uses two different methods of extracting attributes from the product description field.

The first method uses a new (in EDQ 12.2.1.3) Extract Attributes processor with some regular expressions that match patterns of characters in the description field in order to extract the dimensions and material specifications of the screws.

The second method uses a Parse processor, and reference data lists of 'literal' values created by profiling the input description field, to extract new attributes.

- First, click on the **Expand and Standardize Abbreviations** processor, and drill down on the 107 records that have been transformed by it to see the changes it has made to the Description field:

Original Description.WhitespaceNormalized	Original Description.AllReplaced
1/4-20 X 3/8 RH SL MACHINE SCREW 18-8 SS	1/4-20 X 3/8 PAN HEAD SLOTTED MACHINE SCREW 18-8 STAINLESS STEEL
1/4-20 X 5/8 CUP PT SOCKET SET SCR 316 SS	1/4-20 X 5/8 CUP POINT SOCKET SET SCREW 316 STAINLESS STEEL
1/4-20X 1/2" FH SL MACHINE SCR NAT NYLON	1/4-20X 1/2" FLAT HEAD SLOTTED MACHINE SCREW NAT NYLON
1/4-20X1 3/4 HX HD CAP SCREW ALUMINUM	1/4-20X1 3/4 HEX HEAD CAP SCREW ALUMINUM
1/4-20X1 ALLY 25/PK CONE PT SET SCREW	1/4-20X1 ALLOY 25/PK CONE POINT SET SCREW
1/4-20X1/2 ALLOY 50/PK KNURL PT SET SCREW	1/4-20X1/2 ALLOY 50/PK KNURL POINT SET SCREW
1/4-20X2.50 W/NYL PTCH ALLOY SKT HD CAP SCRW	1/4-20X2.50 W/NYL PTCH ALLOY SOCKET HEAD CAP SCREW
1/4-20X3/4" FH TORX DR MACHINE SCREW ZINC PL	1/4-20X3/4" FLAT HEAD TORX DR MACHINE SCREW ZINC PL
1/4-20X3-1/2 ALLOY STL SOCKET HEAD CAP SCREW	1/4-20X3-1/2 ALLOY STL SOCKET HEAD CAP SCREW
1/4-20X6 ZP RND HD PHIL MACHINE SCREW	1/4-20X6 ZP RND HEAD PHIL MACHINE SCREW
1/4-20X7/8 RH PHL MACHINE SCREW ZINC PL	1/4-20X7/8 PAN HEAD PHL MACHINE SCREW ZINC PL
1/4-28 X 1/2 W/PATCH SOC SET SCR ALLOY ST	1/4-28 X 1/2 W/PATCH SOCKET SET SCREW ALLOY ST
1/4-28 X 1/4 W/PATCH SOC SET SCR ALLOY ST	1/4-28 X 1/4 W/PATCH SOCKET SET SCREW ALLOY ST

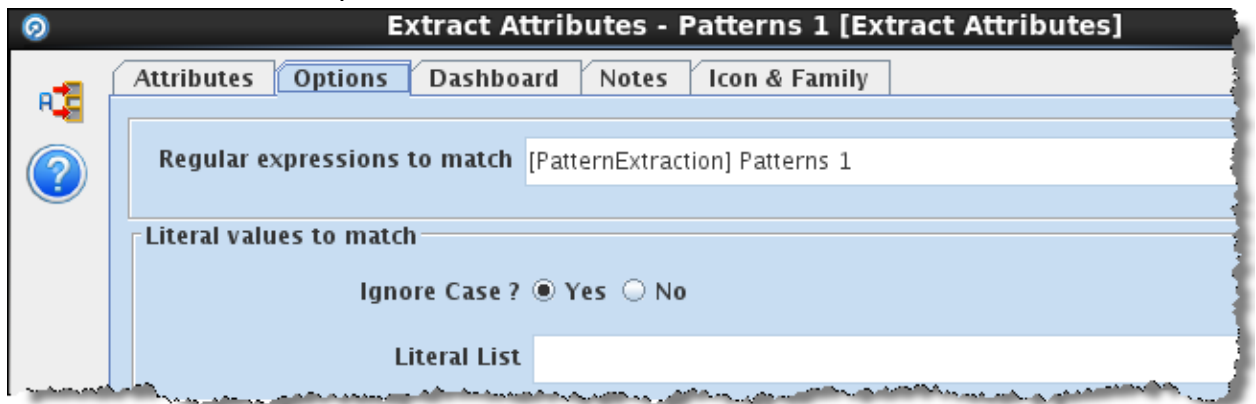
Note that this is an instance of the standard **Replace All** processor, performing simple replacement of delimited tokens in the input string, such as replacing the token SL with SLOTTED and PT with POINT. It can be useful to do this first so that smaller classification lists can be used when extracting literal values in later processing.


- Now, click on the next processor in the chain – **Extract Attributes – Patterns 1** and drill down on the 106 records where it performed at least one extraction:

Original Description.WhitespaceNormalized	AttributeArray.Patterns1	ValueArray.Patterns1
1/4-20 X 3/8 RH SL MACHINE SCREW 18-8 SS	{Dimensions}{MaterialSpec}	{1/4-20 X 3/8}{18-8 SS}
1/4-20 X 5/8 CUP PT SOCKET SET SCR 316 SS	{Dimensions}{MaterialSpec}	{1/4-20 X 5/8}{316 SS}
1/4-20X 1/2" FH SL MACHINE SCR NAT NYLON	{Dimensions}	{1/4-20X 1/2"}
1/4-20X1 3/4 HX HD CAP SCREW ALUMINUM	{Dimensions}	{1/4-20X1 3/4}
1/4-20X1 ALLY 25/PK CONE PT SET SCREW	{Dimensions}{Packaging}	{1/4-20X1}{25/PK}
1/4-20X1/2 ALLOY 50/PK KNURL PT SET SCREW	{Dimensions}{Packaging}	{1/4-20X1/2}{50/PK}
1/4-20X2.50 W/NYL PTCH ALLOY SKT HD CAP SCR W	{Dimensions}	{1/4-20X2.50}
1/4-20X3/4" FH TORX DR MACHINE SCREW ZINC PL	{Dimensions}	{1/4-20X3/4"}
1/4-20X3-1/2 ALLOY STL SOCKET HEAD CAP SCREW	{Dimensions}	{1/4-20X3-1/2}
1/4-20X6 ZP RND HD PHIL MACHINE SCREW	{Dimensions}	{1/4-20X6}
1/4-20X7/8 RH PHL MACHINE SCREW ZINC PL	{Dimensions}	{1/4-20X7/8}
1/4-28 X 1/2 W/PATCH SOC SET SCR ALLOY ST	{Dimensions}	{1/4-28 X 1/2}
1/4-28 X 1/4 W/PATCH SOC SET SCR ALLOY ST	{Dimensions}	{1/4-28 X 1/4}
1/4-28 X 5/16 CUP PT. UNBRAKO SOCKET SET SCREW	{Dimensions}	{1/4-28 X 5/16}
1/4-28X1-3/4 10/PK FLAT HEAD CAP SCREW	{Dimensions}{Packaging}	{1/4-28X1-3/4}{10/PK}
10/24 X 1 18-8 SS NICK LOW-HEAD SKT HD CAP SCW	{MaterialSpec}	{18-8 SS}
10-24 X 1 1/4 RH SL MACHINE SCREW 18-8 SS	{Dimensions}{MaterialSpec}	{10-24 X 1 1/4}{18-8 SS}

Note that this processor extracts attribute values from the input description and creates two array outputs - one array of named attributes, and a corresponding array of values. This output data can then be extracted into separate attributes (Dimensions, Packaging etc.), or made into Entity-Attribute-Value output, with a record for each attribute name-value pair. The output format used will depend on the requirements of the system we will be writing to.

- To inspect how the processor did this, we need to examine its configuration. Double-click on the processor and switch to the Options tab:



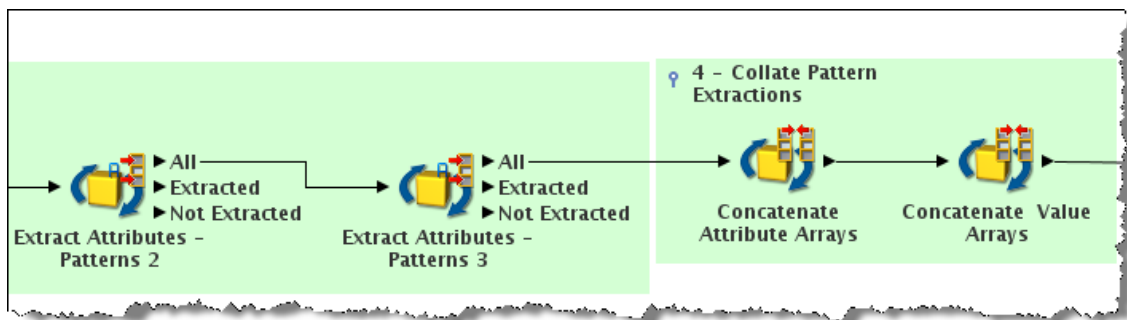
- Click on the  button to have a look at the list of regular expressions that has been used to extract data:

Reference Data Editor - [PatternExtraction] Patterns 1

Viewing page 1 of 1 (8 total saved records)

Pattern	Label	Comment
\d+ \- \d+ \ / \d+ V	Voltage	AC voltage range
\d+ \ / \d+ \- \d+ V	Voltage	AC voltage range
\b(\d+ \-)?\d+ ?SS\b	MaterialSpec	Stainless steel gauge
#\d+ ?X ?\d+ (\ / \d+)?	Dimensions	Screw gage
\bM(\d+ \ /)?\d+ ?X ?\d*(([\.] \d+)?(MM)?(?\d*(([\.] \d+)?)?	Dimensions	Socket head cap screw coding
\b(\d+)?(\d+ \ /)?\d+ ?\ - ?\d+ \ ? ?X ?\d*(([\ -]? \d+ ([\.] \ /)? \d+)?...	Dimensions	Screw spec
\b(\d+ ([\.] \ /)?\d+ ?X ?\d*(([\ -]? \d+ ([\.] \ /)? \d+)?(?\d*(([\ -]?...	Dimensions	Screw spec (short)
\d+ \ / PK	Packaging	Package count

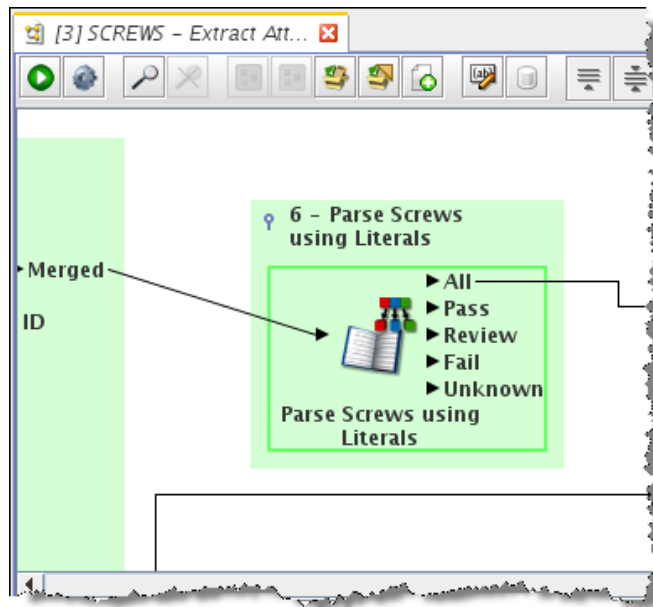
- Here, regular expressions have been used to extract unambiguous patterns of digits (\d), optional punctuation characters (such as hyphens), and indicators of units of measure such as MM to recognize the part of the input string that specifies the Dimensions and Material Specifications (MaterialSpec) of each screw. A simple regular expression (digits followed by PK) has been used to recognize a Packaging attribute.
- The Extract Attributes processor works by extracting data from its input attribute and passing on the remaining input (the parts of the input string where no extractions took place) to the next processor in the chain. This allows multiple steps of extraction to take place on a given category of item, and makes it easier to handle potential ambiguities or overlaps when matching data to regular expressions. We can then easily collate the extractions at the end of the process to create our required output.
- In this process, there are two further Extract Attributes instances, using different regular expressions on the remaining parts of the description. These are simply called **Extract Attributes – Patterns 2** and **Extract Attributes – Patterns 3**. The extracted data is then collated using two instances of the Concatenate Attribute Arrays processor:



- In this case, we want to combine this method of extraction with using a Parse processor, so for output consistency, in stage 5 of the process, we flatten these collated arrays into standard attributes (Dimensions, MaterialSpec etc.) by splitting the arrays and then recombining the records using Group and Merge. We can then more easily combine these attributes with those extracted by our next processor – the Parser.

Note: In many cases with your data, only one of these methods will be needed for a given category (or categories) of product data. The Parse processor supports rich attribute extraction and classification including the use of regular expressions, and Extract Attributes supports the use of literal values. However, we have combined the two approaches here to highlight the flexibility of EDQ, and in order to show that multi-step processing is often the best way to ensure no ambiguities when extracting data. The Parse processor can also standardize literal values as it extracts them, which is useful for our data since we know it contains some values that require this.

- To understand the parse processor, scroll the Canvas to the right and find group 6. Click on the processor named **Parse Screws using Literals**:

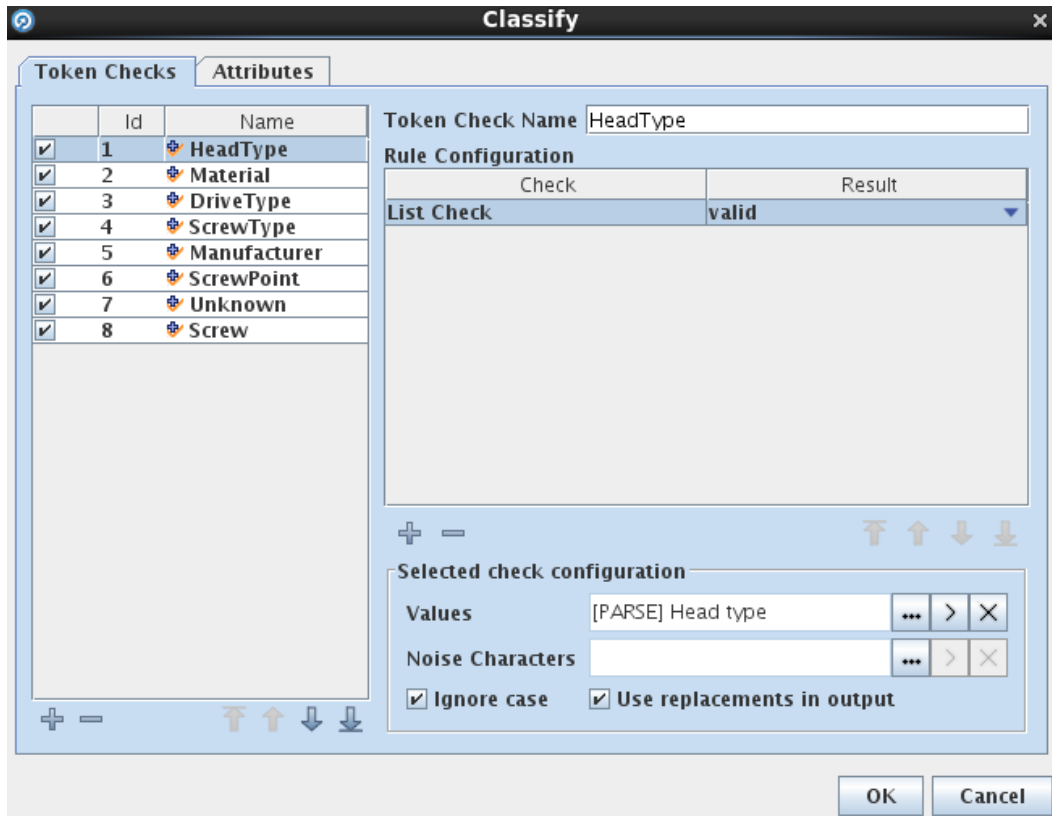


In the Results Browser, switch to the **Data** view so that you can see what this processor has done:

Id	Original Description.AllReplaced	Category	UnclassifiedData	HeadType	Material	D
1526	1/4-20 X 3/8 PAN HEAD SLOTTED MACHINE SCREW ...	SCREW	1,/4,-,20,X,3,/8,18,-,8	PAN HEAD	STAINLESS STEEL	SI
1588	1/4-20 X 5/8 CUP POINT SOCKET SET SCREW 316 ST...	SCREW	1,/4,-,20,X,5,/8,316		STAINLESS STEEL	
1589	1/4-20X 1/2" FLAT HEAD SLOTTED MACHINE SCREW ...	SCREW	1,/4,-,20,X,1,/2,"NAT,NYLON	FLAT HEAD		SI
1592	1/4-20X1 ALLOY 25/PK CONE POINT SET SCREW	SCREW	1,/4,-,20,X,1,25,/PK		ALLOY	
1611	1/4-20X1/2 ALLOY 50/PK KNURL POINT SET SCREW	SCREW	1,/4,-,20,X,1,/2,50,/PK		ALLOY	
1619	1/4-20X3/4" FLAT HEAD TORX DR MACHINE SCREW ...	SCREW	1,/4,-,20,X,3,/4,"DR	FLAT HEAD	ZINC PLATED	TI
1718	1/4-20X6 ZP RND HEAD PHIL MACHINE SCREW	SCREW	1,/4,-,20,X,6	ROUND HEAD	ZINC PLATED	PI
1719	1/4-20X7/8 PAN HEAD PHL MACHINE SCREW ZINC PL	SCREW	1,/4,-,20,X,7,/8,PHL	PAN HEAD	ZINC PLATED	

Here we can see that the Parse processor has extracted and standardized various attributes of screws from the input description, such as **HeadType**, **Material**, **DriveType** and **ScrewType**.

- To understand how the Parse processor has done this, double-click on the processor to open it, and then double-click on the **Classify** sub-processor:



Here, we can see the classification rules the Parse processor has used to classify and standardize data. Note the option to **Use replacements in output** is selected. This means the processor will match values based on the first column in the reference data set used, but will output the corresponding data in the second column. For the HeadType token check, this means 'BTN HEAD' will be standardized to 'BUTTON HEAD' and 'RND HEAD' to 'ROUND HEAD'. Click on the button to see the reference data set used:

Match Value	Standard	Comment	State	Modified By	Modified On
PAN HEAD	PAN HEAD		Active	dnadmin	Jun 28, 2016 6:10:09 AM
FLAT HEAD	FLAT HEAD		Active	dnadmin	Jun 28, 2016 6:10:19 AM
HEX HEAD	HEX HEAD		Active	dnadmin	Jun 28, 2016 6:14:54 AM
LOW HEAD	LOW HEAD		Active	dnadmin	Jun 28, 2016 6:14:45 AM
SOCKET HEAD	SOCKET HEAD		Active	dnadmin	Jul 6, 2016 10:47:36 AM
RND HEAD	ROUND HEAD		Active	dnadmin	Jul 6, 2016 10:51:00 AM
BTN HEAD	BUTTON HEAD		Active	dnadmin	Jul 6, 2016 11:20:51 AM
BUTTON HEAD	BUTTON HEAD		Active	dnadmin	Jul 6, 2016 11:20:58 AM
SQUARE HEAD	SQUARE HEAD		Active	dnadmin	Jul 7, 2016 9:02:38 AM

11. After the parser, the remaining parts of the process show different ways in which we can prepare the extracted attributes into output formats. In this case, four different formats

have been prepared. To examine these formats, click on the Writers at the end of the process chain and examine the data in the Results Browser.

Note: In this process, four different formats have been output (with four different writers). See below for an explanation of each output format:

A) Named Attribute Format

In this writer, each record representing a Screw has been output with a simple named attribute for each piece of data we have extracted from the description (Category, Manufacturer, HeadType, Material etc.):

Id	Original Description	Description Standardized	Category	Manufacturer	HeadType	Material
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	SCREW		SOCKET HEAD	
1010	0-80 X 3/16 CUP PT SOC SET SCR 18-8 SS	0-80 X 3/16 CUP POINT SOCKET SET SCREW 18-8 ST...	SCREW			STAINLESS ST
1048	1 1/4-7X5 1/2 HX HD CAP SCREW-GR 5 ZINC PL	1 1/4-7X5 1/2 HEX HEAD CAP SCREW-GR 5 ZINC PL	SCREW		HEX HEAD	ZINC PLATED
1049	1 1/8-12X4 HX HD CAP SCR-GR 8 ZINC PL(LD)	1 1/8-12X4 HEX HEAD CAP SCR-GR 8 ZINC PL(LD)	SCREW		HEX HEAD	ZINC PLATED
1136	1/2-13 X 16 NON-STD ALLOY SCKT CAP SCR	1/2-13 X 16 NON-STD ALLOY SOCKET CAP SCREW	SCREW			ALLOY

B) Generic Attribute Format

The second writer writes the same records, but outputs the additional attributes in a generic format that will work for many categories of product, with attribute labels alongside their values:

Category	Attribute1	Value1	Attribute2	Value2	Attribute3	Value3	Attribute4	Value4	Attribute5	Value5
SCREW	Manufacturer		HeadType	SOCKET HEAD	Material	STAINLESS STEEL	DriveType		ScrewType	CAP
SCREW 18-8 ST...	Manufacturer		HeadType		Material	ZINC PLATED	DriveType		ScrewType	SOCKET SET
-GR 5 ZINC PL	Manufacturer		HeadType	HEX HEAD	Material	ZINC PLATED	DriveType		ScrewType	CAP
ZINC PL(LD)	Manufacturer		HeadType	HEX HEAD	Material	ZINC PLATED	DriveType		ScrewType	CAP

C) Entity-Attribute-Value format, including Attributes with no value

In the third writer, rather than write a single record for each product id, a record is written for each attribute-value pair, with the id of the product also output. This is then suitable for loading into a target that uses name-value pairs to store product attributes:

Results Browser

Job: [3] SCREWS - Extract Attributes

Id	Original Description	Standardized Description	AttributeName	AttributeValue
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	Grade	
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	Manufacturer	
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	HeadType	SOCKET HEAD
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	Material	
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	DriveType	
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	ScrewType	CAP
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	ScrewPoint	
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	Dimensions	0-80X1 18
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	DIN	
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	MaterialSpec	18-8SS
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	Packaging	

D) Entity-Attribute-Value format, excluding Attributes with no value

The final writer uses the same format as C, but uses a simple No Data Check processor so that records are only written for attribute-value pairs that actually have an extracted value. This is because it is often not necessary to output records for blank values:

Results Browser

Job: [3] SCREWS - Extract Attributes

Id	Original Description	Standardized Description	AttributeName	AttributeValue
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	HeadType	SOCKET HEAD
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	ScrewType	CAP
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	Dimensions	0-80X1 18
1046	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	0-80X1 18-8SS SOCKET HEAD CAP SCREWS	MaterialSpec	18-8SS
1010	0-80 X 3/16 CUP PT SOC SET SCR 18-8 SS	0-80 X 3/16 CUP POINT SOCKET SET SCREW 18-8 ST...	Material	STAINLESS STEEL