



Query_xml, PFF & XSL Transform

Contents

1	Background.....	2
2	Query & PFF	3
3	Query_xml.....	5
4	XSL Transform.....	6
5	XSL Code.....	7

PROSPECT

P L M

1 Background

I have a requirement to create missing forms for a number of revisions. I will export using query_xml, format the results using a PFF. These will then have to be transformed again using an XSL and then these results I will upload using IPS.

If you don't have IPS, then you could change the XSL to output in XML and use either plmxml or tcxl to upload your changes.

2 Query & PFF

The query I'm running is for all items which don't have a form attached using relation IE4_Docs:

Name: RB Missing Documentation - Item

Description:

Query Type: Local Query

Search Class: IE4_ItemRevision

Modifiable Query Types: Local Query

Show Indented Results

Revision Rule:

Attribute Selection

IE4_ItemRevision

pid

	Attribute	User Entry L1...	User Entry Na...		Default Value
	Form:IE4_Docs			IS_NULL	
AND	active_seq	active_seq	active_seq	!=	0

I have created a PFF which just displays Item ID & Revision:

Name: ItemID_RevID

Description:

Principle Query Object: ItemRevision

Add Clauses From: ItemRevision

Property Selection

ItemRevision

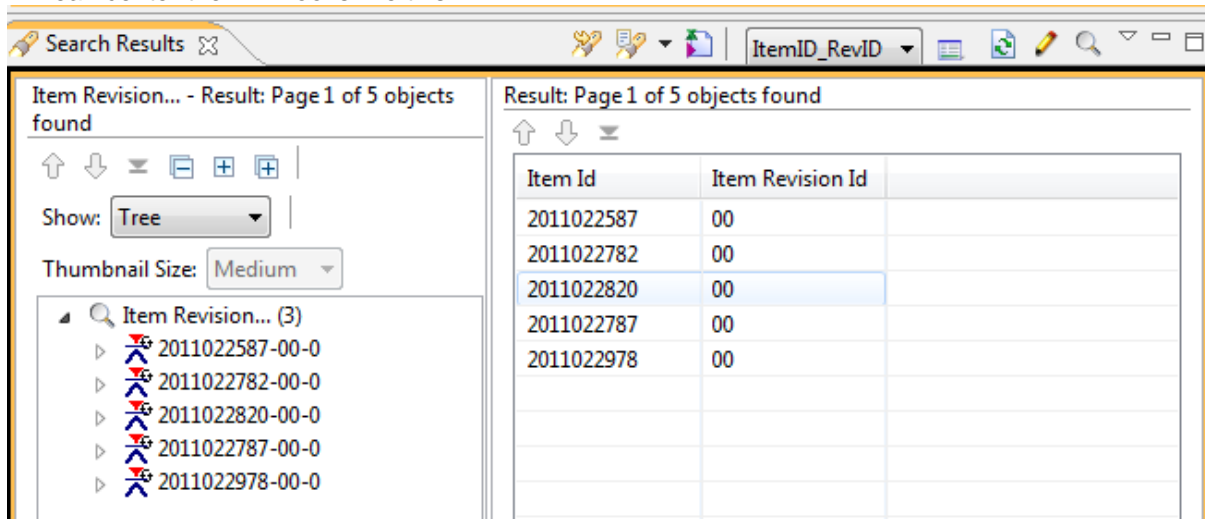
- Application Name
- Application Relation
- Application Version
- BOM View Revisions [PSBOMViewRevision]

Property Path	Column Names
ItemRevision.items_tag.item_id	Item Id
ItemRevision.item_revision_id	Item Revision Id

Create Delete Modify Clear

PROSPECT P L M

In Teamcenter the PFF looks like this:



The screenshot shows the Teamcenter Search Results interface. The search criteria is 'ItemID_RevID'. The results are displayed in a table with the following data:

Item Id	Item Revision Id
2011022587	00
2011022782	00
2011022820	00
2011022787	00
2011022978	00

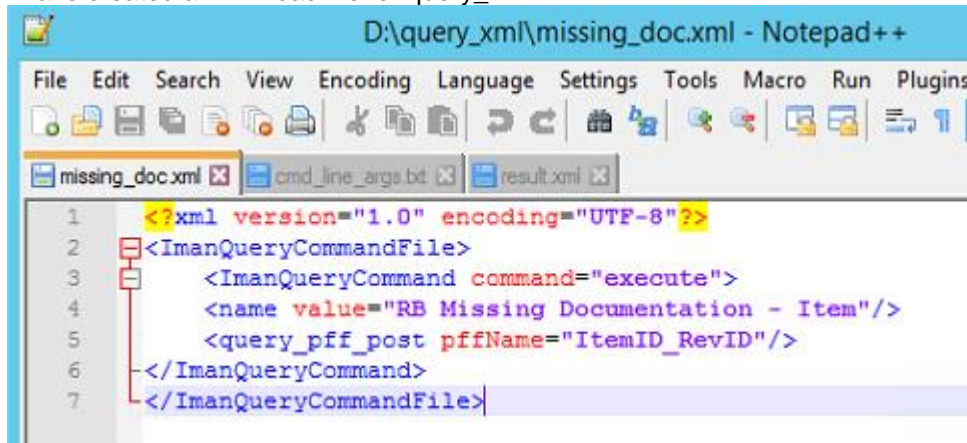
The left sidebar shows a tree view of the search results, with the following items listed:

- Item Revision... (3)
 - 2011022587-00-0
 - 2011022782-00-0
 - 2011022820-00-0
 - 2011022787-00-0
 - 2011022978-00-0

PROSPECT

3 Query_xml

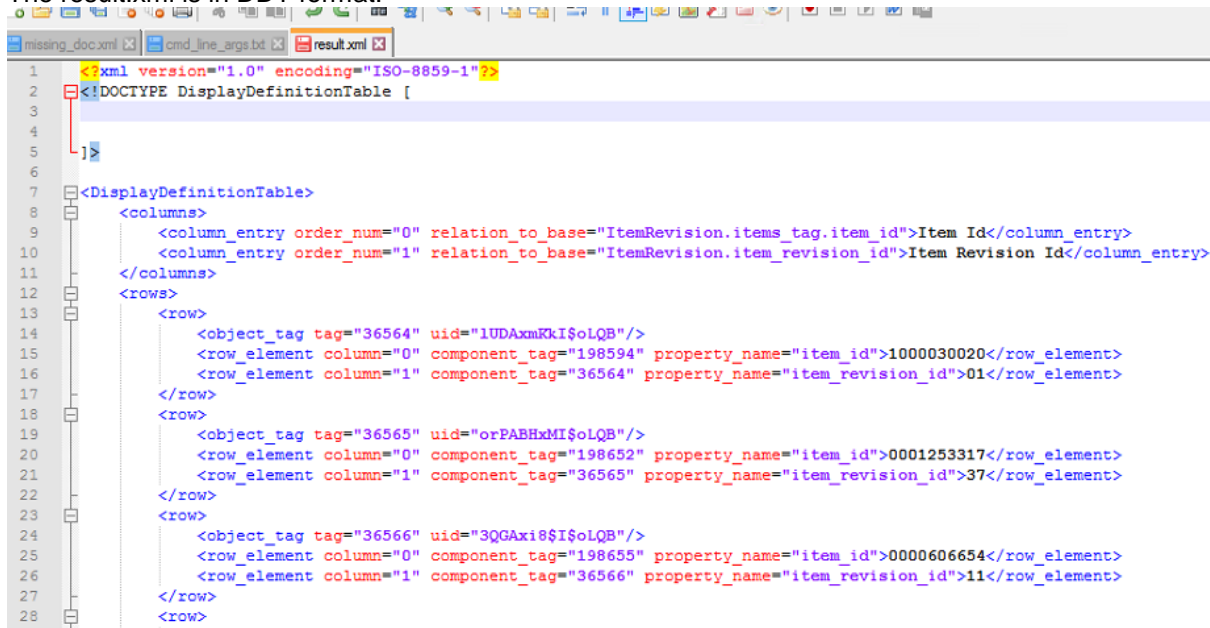
I have created an xml load file for query_xml:



```
<?xml version="1.0" encoding="UTF-8"?>
<ImanQueryCommandFile>
  <ImanQueryCommand command="execute">
    <name value="RB Missing Documentation - Item"/>
    <query_pff_post pffName="ItemID_RevID"/>
  </ImanQueryCommand>
</ImanQueryCommandFile>
```

I then run the following cmd from a TC cmd prompt:
query_xml -f=missing_doc.xml -u= -p= -o=result.xml

The result.xml is in DDT format:




```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE DisplayDefinitionTable [
]
<DisplayDefinitionTable>
  <columns>
    <column_entry order_num="0" relation_to_base="ItemRevision.items_tag.item_id">Item Id</column_entry>
    <column_entry order_num="1" relation_to_base="ItemRevision.item_revision_id">Item Revision Id</column_entry>
  </columns>
  <rows>
    <row>
      <object_tag tag="36564" uid="1UDAXmKkI$olQB"/>
      <row_element column="0" component_tag="198594" property_name="item_id">1000030020</row_element>
      <row_element column="1" component_tag="36564" property_name="item_revision_id">01</row_element>
    </row>
    <row>
      <object_tag tag="36565" uid="orPABHxMI$olQB"/>
      <row_element column="0" component_tag="198652" property_name="item_id">0001253317</row_element>
      <row_element column="1" component_tag="36565" property_name="item_revision_id">37</row_element>
    </row>
    <row>
      <object_tag tag="36566" uid="3QGAXi8$I$olQB"/>
      <row_element column="0" component_tag="198655" property_name="item_id">0000606654</row_element>
      <row_element column="1" component_tag="36566" property_name="item_revision_id">11</row_element>
    </row>
  </rows>
</DisplayDefinitionTable>
```

4 XSL Transform

I need this in a format I can upload to Teamcenter. I use IPS so the format needs to be delimited: Itemid~revid

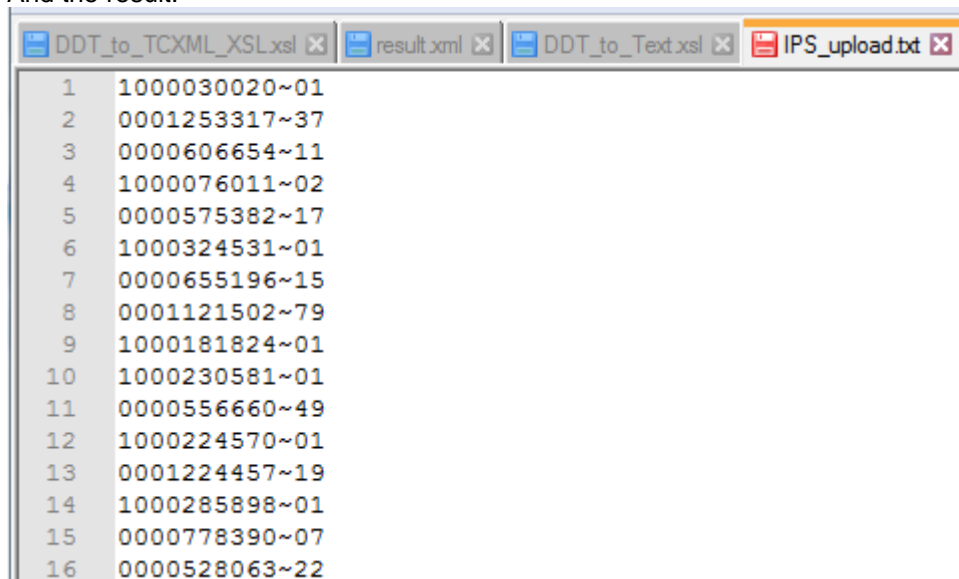
I use an XSL which I can use to transform the xml (I use a free product called X-RAY, but there are others).

My xsl (code is at end so you can copy it)



```
1 <?xml version="1.0" ?>
2 <xsl:stylesheet version="1.0"
3   xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
4   xmlns:plm="http://www.plmxml.org/Schemas/PLMXMLSchema"
5   xmlns:crf="http://ExternalFunction.setFile"
6   xmlns:msxsl="http://exslt.org/common"
7   xmlns:str="http://exslt.org/strings">
8   <xsl:output method="text" omit-xml-declaration="yes" indent="no"/>
9   <xsl:template match="/*">
10     <xsl:for-each select="//row[./row_element]">
11       <xsl:call-template name="Values">
12         </xsl:call-template>
13     </xsl:for-each>
14   </xsl:template>
15
16   <xsl:template name="Values" match="text()">
17     <xsl:variable name="column0" select="row_element[@column='0']/../text()"/>
18     <xsl:variable name="column1" select="row_element[@column='1']/../text()"/>
19     <xsl:value-of select="$column0"/>
20     <xsl:text>~</xsl:text>
21     <xsl:value-of select="$column1"/>
22     <xsl:text>~</xsl:text>
23   </xsl:template>
24 </xsl:stylesheet>
```

And the result:



```
1 1000030020~01
2 0001253317~37
3 0000606654~11
4 1000076011~02
5 0000575382~17
6 1000324531~01
7 0000655196~15
8 0001121502~79
9 1000181824~01
10 1000230581~01
11 0000556660~49
12 1000224570~01
13 0001224457~19
14 1000285898~01
15 0000778390~07
16 0000528063~22
```

5 XSL Code

```
<?xml version="1.0" ?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:plm="http://www.plmxml.org/Schemas/PLMXMLSchema"
  xmlns:crf="http://ExternalFunction.setFile"
  xmlns:msxsl="http://exslt.org/common"
  xmlns:str="http://exslt.org/strings">
  <xsl:output method="text" omit-xml-declaration="yes" indent="no"/>
<xsl:template match="/*">
  <xsl:for-each select="//row[./row_element]">
    <xsl:call-template name="Values">
      </xsl:call-template>
    </xsl:for-each>
</xsl:template>
<xsl:template name="Values" match="text()">
  <xsl:variable name="column0" select="row_element[@column='0']/./text()"/>
  <xsl:variable name="column1" select="row_element[@column='1']/./text()"/>
  <xsl:value-of select="$column0"/>
  <xsl:text>~</xsl:text>
  <xsl:value-of select="$column1"/>
  <xsl:text>~&#10;</xsl:text>
</xsl:template>
</xsl:stylesheet>
```