

Oracle® Agile Product Lifecycle Management for Process Search Extensibility Guide

Feature Pack 4.2

E66824-01

April 2016

ORACLE®

Copyrights and Trademarks

Agile Product Lifecycle Management for Process

Copyright © 1995, 2016 Oracle and/or its affiliates. All rights reserved.

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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software 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

PREFACE	5
Audience	5
Variability of Installations	5
Documentation Accessibility.....	5
Access to Oracle Support	5
Software Availability	5
CHAPTER 1—OVERVIEW	6
CHAPTER 2—ADJUSTING DEFAULT TOP SEARCH CRITERIA	7
CHAPTER 3—ADJUSTING DISPLAY AND AVAILABLE COLUMNS	9
EQTDisplayColumnsConfig Structure	9
Adjusting Display Columns.....	9
Available Columns.....	12
CHAPTER 4—ADD NEW SEARCH CRITERIA AND DISPLAY COLUMN	13
Example: Add Extended Attribute as a Display Column	14
Example: Add Search Criteria and Display Column which Represents Phone for Facility Profile Search	16

Preface

Audience

This guide is intended for client programmers involved with integrating Oracle Agile Product Lifecycle Management for Process. Information about using Oracle Agile PLM for Process resides in application-specific user guides. Information about administering Oracle Agile PLM for Process resides in the *Oracle Agile Product Lifecycle Management for Process Administrator User Guide*.

Variability of Installations

Descriptions and illustrations of the Agile PLM for Process user interface included in this manual may not match your installation. The user interface of Agile PLM for Process applications and the features included can vary greatly depending on such variables as:

- Which applications your organization has purchased and installed
- Configuration settings that may turn features off or on
- Customization specific to your organization
- Security settings as they apply to the system and your user account

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.

Software Availability

Oracle Software Delivery Cloud (OSDC) provides the latest copy of the core software. Note the core software does not include all patches and hot fixes. Access OSDC at:

<http://edelivery.oracle.com>

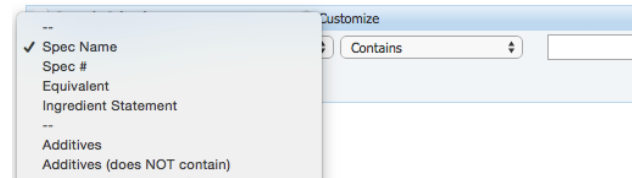
Chapter 1—Overview

While the standard application search behavior allows users to customize their experience individually, search extensibility allows you to extend this behavior. Using search extensibility you can do the following:

- [Chapter 2—Adjusting Default Top Search Criteria](#)
- [Chapter 3— Adjusting Display and Available Columns](#)
- [Chapter 4— Add New Search Criteria and Display Column](#)

Chapter 2—Adjusting Default Top Search Criteria

The default search criteria and the items displayed at the top of the criteria dropdown list are configurable. For example, the standard behavior displays Spec Name as the default criterion. This can be changed to any available criteria like Spec # or Equivalent.



Top Criteria is adjustable using the EQTSearchablePropertyConfigs.xml file inside the config\Extensions folder. The following steps can be followed to adjust any search in PLM4P.

1. Backup the EQTSearchablePropertyConfigs.xml file.
2. Find the Model node by the value of attribute modelName in this file. This node represents the primary search focus. For example, for Formulation Specs the model name is “Formulation Specifications”. Generally this name is easily recognizable, however you can find the modal name in the UI inside the search Customize window as part of the apply button. For example, “Apply to all [Model Name] Searches”.
3. Change the values of corresponding SearchableProperty attributes to change search criteria.
 - To add new criteria, add a new SearchableProperty node with the preferred PropertyID.
 - Remove the SearchableProperty nodes or change enabled to “false” to display the criteria further in the list.
4. Reset IIS to review the changes in the search page.

The Model node structure in EQTSearchablePropertyConfigs looks like the following:

```
<Model modelName="Formulation Specifications">
  <SearchableProperty propertyId="SpecName" sequence="1" enabled="true" comment=""></SearchableProperty>
  <SearchableProperty propertyId="TemplateSpecName" sequence="2" enabled="true" comment=""></SearchableProperty>
  <SearchableProperty propertyId="SpecNumber" sequence="3" enabled="true" comment=""></SearchableProperty>
  <SearchableProperty propertyId="TemplateSpecNumber" sequence="4" enabled="true" comment=""></SearchableProperty>
  <SearchableProperty propertyId="SystemEquivalent" sequence="5" enabled="true" comment=""></SearchableProperty>
  <SearchableProperty propertyId="IngredientStatement" sequence="6" enabled="true" comment=""></SearchableProperty>
</Model>
```

<SearchableProperty> contains following attributes:

Attribute	Value type	Description
PropertyId	string	<p>The id of the search criteria. The criteria id can be found by viewing the html source of the search page and finding the search criteria <option> tag. For example, if you wanted Short Name to appear as top criteria you would find the following tag in the html source. The value “ShortName” would be the PropertyID.</p> <pre><option value="ShortName" title="Short Name">Short Name</option></pre> <p>Required</p>
sequence	integer	<p>Sequence used to display the criteria. The default criterion has the sequence of 1. Make sure the sequence is unique.</p> <p>Required</p>
enabled	boolean	<p>To remove a criteria set this to false or remove the entire node.</p> <p>Required</p>
comment	string	<p>For reference only.</p> <p>Optional</p>

Chapter 3—Adjusting Display and Available Columns

You can adjust which columns are displayed by default for each search view location as well as the default sorting behavior. You can also adjust the columns that are available to users during the UI customize process. All adjustments are made using the EQTDisplayColumnsConfig.xml file inside the config\Extensions folder.




EQTDisplayColumnsConfig Structure

The EQTDisplayColumns node structure looks like the following:

```
<EQTDisplayColumns>
  <Model modelName="[MODEL_NAME]">
    <!-- Default display columns definition, required -->
    <DisplayColumns location="default">
      <!-- One or more column definition-->
      <ColumnInfo/>
      <ColumnInfo/>
      .....
    </DisplayColumns>
    <!-- Display columns definition for specific location, optional -->
    <DisplayColumns location="[LOCATION_ID]">
      <!-- One or more column definition-->
      <ColumnInfo/>
      <ColumnInfo/>
      .....
    </DisplayColumns>
    <!-- Available columns definition, required -->
    <AvailableColumns>
      <!-- One or more column definition-->
      <ColumnInfo/>
      <ColumnInfo/>
      .....
    </AvailableColumns>
  </Model>
</EQTDisplayColumns>
```

Adjusting Display Columns

Every search view location comes standard with default search result columns displayed. For example, the main Material Specification search shows the following columns by default.

Search Results		Export							
	Spec #	Spec Name	Short Name	Type	Status	Category	Supersedes	Preferred Equivalent	
	5077415-001	Granulated Sugar (Sucrose)		Raw Material	Official	Sweeteners » Sugar » Dry	017581 - 12/11/2003		▼
	5077415-002	Granulated Sugar (Sucrose)		Raw Material	Draft	Sweeteners » Sugar » Dry	5077415-001 - Granulated Sugar (Sucrose)		▼
	5077422-001	Brown Sugar - Light		Raw Material	Draft	Sweeteners » Sugar » Dry	34481 - 07/19/1994		▼

While individual users can configure these columns for themselves you may want to change the standard display columns to better fit the larger user community.

Default display columns are adjustable for every unique UI location using the EQTDisplayColumnsConfig.xml file inside the config\Extensions folder. The following steps can be followed to adjust any search in PLM4P.

1. Back up the EQTDisplayColumnsConfig.xml file.
2. Find the Model node by the value of attribute modelName in this file. This node represents the primary search focus. For example, for Material Specs the model name is “Material Specifications”. Generally this name is easily recognizable, however you can find the modal name in the UI inside the search Customize window as part of the apply button. For example, “Apply to all [Model Name] Searches”.
3. Find the DisplayColumns node for the specific Location ID you are interested in.
4. Change the values of corresponding ColumnInfo attributes under the DisplayColumns node.
 - To add a new display column, add a new ColumnInfo node.
 - Remove the ColumnInfo node to remove the default display column.
5. Reset IIS to review the changes in the search page.

<DisplayColumns> contains following attributes:

Attribute	Description
location	<p>The ID of the specific UI location where you want to set specific display columns. Every UI location has a unique ID. For example, you can show different columns on the main material specification search versus the material specification search popup that appears on the formulation specification when adding inputs.</p> <p>The location of “default” applies to all locations not specifically defined. You can create a new DisplayColumns group if you want to specify a location id that has yet to be defined in the configuration file. You can find the location id by opening the Customize window of the search you are interested in and looking below the Results Per Page section. For example, the Main Material Specification search UI location ID is GSM.MAT.Main.</p> <p>The location value should be unique under a Model node. Required.</p>
orderByColumns	<p>The order the column will appear. The sort order rule is defined as [COLUMN_INDEX1] [DIRECTION1],[COLUMN_INDEX2] [DIRECTION2],.....</p> <p>The columns index starts from 0, and the direction value could be asc or desc. If the direction is not specified, it will be treated as asc. If orderByColumns="-", no sort order rule is applied. Required.</p>

Attribute	Description
captionColumn	Identifies the field that is linked in the search export for all apps except for GSM and SCRM. The index of caption column starts from 0. Required.

<ColumnInfo> contains the following attributes:

Attribute	Description
width	This is the column width designated in pixels. Required.
dataField	Represents the sort order rule. For example, orderByColumns="-" means no sort order rule applied, and orderByColumns="0 desc,5" means the results should order by the first column in descend order and the sixth column in ascend order. Required.
dataFieldCaption	The translatable label that will be used as the column header. Required.
isRequired	Whether this column can be removed during user customization. Required.
dataTransformer	The data transformer of the column. Optional.
dataFormatter	The data formatter of the column. Optional.
hideHeader	Whether the column header should be blank in the search result. Optional; default value is false.
sortingEnabled	Whether the sort order of this column can be customized by the user. If sortingEnabled="false", this column will not be listed in the Sort Order popup of the Customize window. Optional; default value is true.
isAggregatable	Whether data in this column can be aggregated. Optional; default value is false.
aggregationDelimiter	The delimiter between the aggregated values. The attribute is effective only if isAggregatable="true". Optional.
enableHandler	Determines whether this column is active. Only active columns will show up in the search result. Optional.
disableExport	Whether the column is included in the search result export. Optional, default value is "false".

Below is an example of the Display Columns configuration for the Main Material Specification Search.

```
<DisplayColumns location="GSM.MAT.Main" orderByColumns="3|desc,0" captionColumn="0">
  <ColumnInfo width="74" dataField="SpecNumber" dataFieldCaption="lblSpecNumber"
provideInSelectJS="true" isRequired="true"></ColumnInfo>
  <ColumnInfo width="168" dataField="SpecName" dataFieldCaption="lblSpecName"
isRequired="false"></ColumnInfo>
  <ColumnInfo width="74" dataField="ShortName" dataFieldCaption="lblSpecShortName"
isRequired="false"
enableHandler="Class:Xeno.Prodika.EQTService.SearchablePropertyHandlers.FeaturePropertyItem
EnabledResolverFactory,EQTService$GSM.EQT.SpecSummary.MaterialSpec.ShortName.Enabled"></Col
umnInfo>
</DisplayColumns>
```

Available Columns

You can also remove or add columns available to users during the customize process. For example, let's say you want to add Extended Attribute "Brix" as a Display Column; however it's not something that all users will appreciate as a default column. You will want to add this column as just an Available Column to users so they can choose when to use it as a default Display Column. To adjust available columns, you follow the same approach as above however all available columns are defined in the <AvailableColumns> node below the <DisplayColumns> nodes. Available Columns are configured per Model and cannot be unique per location ID.

Chapter 4—Add New Search Criteria and Display Column

You can add new search criteria to existing search models. You can also add new display columns to existing search views. All additions are made using the EQTModelCustomExtensions.xml located inside config\Extensions folder.

The following steps should be followed:

1. Back up the EQTModelCustomExtensions.xml file.
2. Find or add the ExtendEntityModel node for the model you would like to extend.
3. Add the EQT Property Node (TextProperty, NumericProperty, DateTimeProperty or PickListProperty)
4. Add a TableAlias if the table is used more than once across a single search model.
5. If adding a column, follow the steps in “Adjusting Display and Available Columns” section to add your new column using the EQT Property you added above.
6. Reset IIS to review the changes in the search page.

The following property nodes are available:

Property node name	Description
NumericProperty	Numeric property
TextProperty	Text property
DateTimeProperty	Date property
PickListProperty	Pick list, usually for searching

Property nodes contain the following attributes:

Attribute	Description
caption	The label shown as criteria name and/or column header. You can use an extended translation or free text. Required
name	The id of the property. Required
isMultiValued	For PickListProperty Only. This determines whether the pick list returns multiple values. [Yes, No]
forSearching	Whether this property will appear as search criteria. [Yes, No]
isTranslateable	Whether the field returned is language aware. [Yes, No]
isUsePrimaryLang	Whether EQT uses the primary language for searching. The primary language is the UI Language on User Profile and Preferences. If set to false, the secondary language which is

Attribute	Description
	the Free Text Language on User Profile and Preferences will be used for searching.
isUnicode	Whether the field returned is Unicode. [Yes, No]
pickListType	For PickListProperty Only. Pick list type, should be S or M or H, which mean Single Column List or Multi Column List or Hierarchy
isFreeTextLanguage	Whether the translatable field returned is free text. [Yes, No]

Property nodes contain the following sub nodes:

Sub node name	Description
<u>FieldName</u>	The value field name. Should be TableName.ColumnName Required
JoinLinks\Links	Equation for table join, like Table1.Column1 = Table2.Column *= means left outer join =* means right outer join
TextCriterionOperations	For TextProperty Only. These are the search criteria operators: startsWith, endsWith, contains, equals, notEquals Required
CriterionOperations	These are the search criteria operators for all other property types: greaterThan, lessThan, equals Required
DataSourceName	For PickListProperty Only. This is the datasource used for the picklist. Required
ConditionalExpressions	Allows you to provide additional filters to your search query. Optional

Example: Add Extended Attribute as a Display Column

You are able to add numeric, quantitative range and Boolean extended attributes as display columns using the following out of the box EA Type Transformers.

Available Extended Attribute Transformers

EA Type	Transformer
<u>Boolean</u>	ExtendedAttributeBooleanDataTransformerFactory\$EA_Boolean_ID
<u>Numeric</u>	ExtendedAttributeNumericDataTransformerFactory\$EA_Numeric_ID
<u>Quantitative Range</u>	ExtendedAttributeQuantitativeRangeDataTransformerFactory\$EA_Quantitative_Range_ID

Below is an example of adding a Quantitative Range EA as a display column.

1. Find node <ExtendEntityModel extends="Packaging Specifications"> in EQTModelCustomExtensions.xml. Add an additional property node under the

ExtendEntityModel node.

```
<TextProperty caption="lblEARoot1" name="EARoot1" forSearching="No" isTranslateable="No"
isFreeTextLanguage="No" isUnicode="No">
    <FieldName>specSummary.SpecID</FieldName>
</TextProperty>
```

2. Find node <Model modelName="Packaging Specifications"> in EQTDisplayColumnsConfig.xml. Add a ColumnInfo node like this under the AvailableColumns node. The data transformer retrieves the Quantitative Range type extended attribute value which ID is "EA_ID".

```
<ColumnInfo width="115" dataField="EARoot1" dataFieldCaption="lblEARoot1"
dataTransformer="Class:Xeno.Prodika.EQTService.DataTransformers.ExtendedAttributeQuantitativeRangeDataTransformerFactory$EA_ID" isRequired="false"></ColumnInfo>
```

3. Add translation value for the data field caption.

To re-use the existing extended attribute name, use the following approach:

```
insert into commonXLAExtensionCacheItem ( pkid, Id, fkParent, langID, Value )
select 'PROVIDE_1059_UNIQUE_PKID', N'PROVIDE_dataFieldCaption', '10587ed83e2e-5fd8-
4eaa-958e-b854d92a7985', ml.langID, ml.Name from
commonExtendedAttributeType tp
inner join commonExtendedAttributeTypeML ml on tp.pkid = ml.fkExtendedAttributeType
where AttributeID = 'EA_ATTRIBUTE_ID';
```

To create a new name, use the following approach:

```
insert into commonXLAExtensionCacheItem ( pkid, fkParent, langID, Id, Value )
SELECT 'PROVIDE_1059_UNIQUE_PKID', '10587ed83e2e-5fd8-4eaa-958e-b854d92a7985', 0,
N'lblEARoot1', N'EA Root1 Translation' FROM DUAL WHERE NOT EXISTS(SELECT 1 FROM
commonXLAExtensionCacheItem WHERE pkid = 'Translation PKID' and langID = 0);
```

Note: Verify that the Translation PKID that is being inserted is unique per language id. It must start with 1059.

4. Reset IIS and review the changes:
 - a. Open the Packaging Specifications Search page, open the Add Columns dialog from the Customization page. Notice the **EA Root1 Translation** column is listed as available column.
 - b. Add the **EA Root1 Translation** column and apply the changes. Run a search. Notice the EA value is returned in the search result.
 - c. Because the new property is not for searching, it is not listed in the Key field drop down list.

Example: Add Search Criteria and Display Column which Represents Phone for Facility Profile Search

1. Find node <ExtendEntityModel extends="Facilities"> in EQTModelCustomExtensions.xml. Add an additional property node under the ExtendEntityModel node.

```
<TextProperty caption="lblFacilityPhone" name="FacilityPhone" forSearching="Yes"
isTranslateable="No" isFreeTextLanguage="No" isUnicode="Yes">
    <FieldName>
        scrmFacility.Phone
    </FieldName>
    <TextCriterionOperations startsWith="On" contains="On"
equals="On"></TextCriterionOperations>
</TextProperty>
```

2. Find node <Model modelName="Facilities"> in EQTDisplayColumnsConfig.xml. Add a ColumnInfo node like this under the AvailableColumns node.

```
<ColumnInfo width="75" dataField="FacilityPhone"
dataFieldCaption="lblFacilityPhone" isRequired="false"></ColumnInfo>
```

3. Add translation value for the data field caption.
*insert into commonXLAExtensionCacheItem (pkid, fkParent, langID, Id, Value)
SELECT 'Translation PKID', '10587ed83e2e-5fd8-4eaa-958e-b854d92a7985', 0,
N'lblFacilityPhone', N'Phone' FROM DUAL WHERE NOT EXISTS(SELECT 1 FROM
commonXLAExtensionCacheItem WHERE pkid = 'Translation PKID' and langID = 0);
Verify that the Translation PKID that is being inserted is unique per language id. It must start
with 1059.*
4. Reset IIS and review the changes in Facility Profile search page.
 - a. User can search for Facility Profile by Phone.
 - b. User can add Phone to the search result.