

Function Registry

Identifier: org.eclipse.bpel.common.model.functionRegistry

Since: 0.3.0

Description: Function registry extension points allows 3rd party plugins to register Meta Information about functions. A function characteristics are that it has a return type and has 0-N arguments. Functions have themselves namespaces and they also live with a container that itself belongs to a namespace as well. This is to distinguish XPath functions from other language functions.

Configuration Markup:

```
<!--ELEMENT extension (registry)-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id    CDATA #IMPLIED
  name  CDATA #IMPLIED
-->
```

```
<!--ELEMENT registry EMPTY-->
<!--ATTLIST registry
  uri          CDATA #REQUIRED
  namespace    CDATA #REQUIRED
-->
```

- uri -
- namespace -

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]

Embedded Editors

Identifier: org.eclipse.bpel.common.ui.embeddedEditors

Since: Since 0.0.1 of the BPEL editor plugin.

Description: This extension point enumerates the text editors which can be used in the "embedded" sense to edit expressions in the expression languages selected by the BPEL process.

Such "embedded" editors differ from a normal text editor in eclipse in that they do not open a resource (such as a file) but operate on some internal memory representation of a document. Such editors also are not openable in the main editor area but rather open in various views where their services are necessary.

Configuration Markup:

```
<!ELEMENT extension (editor+)>
<!ATTLIST extension
  point CDATA #REQUIRED
  id    CDATA #IMPLIED
  name  CDATA #IMPLIED
>

<!ELEMENT editor EMPTY>
<!ATTLIST editor
  class          CDATA #REQUIRED
  contributorClass CDATA #IMPLIED
  id             CDATA #REQUIRED
  name           CDATA #REQUIRED
>
```

- **class** - The name of a class that implements org.eclipse.ui.IEditorPart..
- **contributorClass** - The name of a class that implements org.eclipse.ui.IEditorActionBarContributor. This attribute should only be defined if the class attribute is defined. This class is used to add new actions to the workbench menu and tool bar which reflect the features of the editor type.
- **id** - A unique id that will be used to identify this editor
- **name** - The name of this editor (XPath Editor, Ruby Editor).

Examples:

```
<extension
  point="org.eclipse.bpel.common.ui.embeddedEditors">
  <editor
    contributorClass="org.eclipse.ui.texteditor.BasicTextEditorActionContributor"
    class="org.eclipse.bpel.ui.editors.xpath.XPathTextEditor"
    name="XPath 1.0 Text Editor."
    id="org.eclipse.bpel.ui.editors.xpath.XPathTextEditor">
  </editor>
  <editor
    contributorClass="org.eclipse.ui.texteditor.BasicTextEditorActionContributor"
    class="org.eclipse.bpel.ui.editors.TextEditor"
    name="Default Expression Editor"
    id="org.eclipse.bpel.ui.editors.TextEditor">
  </editor>
</extension>
```

Supplied Implementation: An implementation of an XPath text editor and a Default text editor is provided.

Palette Additions

Identifier: org.eclipse.bpel.common.ui.paletteAdditions

Since: 0.0.1

Description: This extension point allows for contributing BPEL palette items. You can either create an individual addition for each palette element or you can create a bulk addition by specifying a palette provider.

Configuration Markup:

```
<!--ELEMENT extension (additions)-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id     CDATA #IMPLIED
  name   CDATA #IMPLIED
-->

<!--ELEMENT additions (addition*)-->
<!--ATTLIST additions
  targetEditor CDATA #REQUIRED
  provider      CDATA #IMPLIED
  importance    CDATA #IMPLIED
-->
```

Additions represents the palette additions that will be made by this extension. Either **provider** must be set or a list of **addition** childrens must be specified. When setting **provider** it allows for bulk in-code creation of the palette items and categories.

- **targetEditor** - The target editor for the palette additions (the editor id). While this value must be set, it currently does not have any affect other then it must much the BPEL editor id, which is org.eclipse.bpel.ui.bpeleditor.
- **provider** - The provider is a class which implements IPaletteProvider. It allows for "bulk" creation and addition of palette items to the BPEL palette.
- **importance** - An importance assigned to these palette additions. A lower number means a contribution is more important and will be added first. The default value for importance is 10. It is a numeric value. The BPEL palette additions which the editor itself provides are of importance 5.

```
<!--ELEMENT addition EMPTY-->
<!--ATTLIST addition
  class          CDATA #REQUIRED
  category       CDATA #IMPLIED
  default        (true | false)
  categoryId     CDATA #IMPLIED
  categoryOrder  CDATA #IMPLIED
-->
```

A palette addition.

- **class** - A class which extends org.eclipse.gef.palette.CreationToolEntry.
- **category** - The label of the category id that will be used for this category. Note that only the first tool entry with the category id will be used to create a palette container in the palette.
- **default** - Whether this single tool addition is the default tool for the category. Last tool setting this value to **true** within a palette container wins.
- **categoryId** - The categoryid represents an internal category specification. The format of this entry is a dotted identifier, something like `bpel.actions` or `bpel.faults` or `user.macros`. It is used instead of the category as a way of slotting and creating new categories in the palette. The category attribute is the actual label of the category.
- **categoryOrder** - Represents the category order. Default order is 100. The built-in BPEL categories are 10,20,30 respectively and have category ids `bpel.actions`, `bpel.control`, and `bpel.faults`.

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

resourceSetProvider

Identifier: org.eclipse.bpel.common.ui.resourceSetProvider

Since: 0.0.4

Description: This extension point provides for a ResourceSet to the underlying edit model which must be a sub-class of BPELResourceSetImpl.

Configuration Markup:

```
<!--ELEMENT extension (provider)-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id      CDATA #IMPLIED
  name    CDATA #IMPLIED
-->
```

- **point** - The extension point name.
- **id** - Extension point id.
- **name** - Extension point name

```
<!--ELEMENT provider EMPTY-->
<!--ATTLIST provider
  class CDATA #REQUIRED
-->
```

The provider for the resourceSet extension point.

- **class** - The class the implements IResourceSetProvider.

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]

Hover Helpers for BPEL Editor Extension Point

Identifier: org.eclipse.bpel.ui.hoverHelpers

Since: 0.0.1

Description: The hover helper extension point. It is used to present hover help for markers and model objects in the main editor diagram.

Configuration Markup:

```
<!ELEMENT extension (hoverHelper)>
<!ATTLIST extension
  point CDATA #REQUIRED
  id     CDATA #IMPLIED
  name   CDATA #IMPLIED
>
```

The Hover Helper extension point.

- **point** - The extension point name.
- **id** - A unique id assigned to this extension point implementation.
- **name** - The name associated with this extension point implementation.

```
<!ELEMENT hoverHelper EMPTY>
<!ATTLIST hoverHelper
  class CDATA #REQUIRED
>
```

The hover helper extension point element.

- **class** - Class which implements org.eclipse.bpel.ui.IHoverHelper.

Examples: Here is the simple hover helper extension point provided in the BPEL Editor.

```
<extension
  id="org.eclipse.bpel.ui.hover.simple.hover.helper"
  name="Marker Hover Help"
  point="org.eclipse.bpel.ui.hoverHelpers">
  <hoverHelper class="org.eclipse.bpel.ui.hovers.SimpleHoverHelper" />
</extension>
```

API Information: [Enter API information here.]

Supplied Implementation: There is an implementation of this extension point in org.eclipse.bpel.ui plugin. The class which implements this is org.eclipse.bpel.ui.hovers.SimpleHoverHelper.

Expression Language Editors

Identifier: org.eclipse.bpel.ui.expressionEditors

Since: 0.0.1

Description: The BPEL Editor provides a mechanism to plugin expression editors for the various expression languages that may be used by a BPEL runtime. XPath 1.0 is the default standard expression language that is used by BPEL. However, implementations may choose to provide other expression language implementations (such as JavaScript).

Configuration Markup:

```
<!ELEMENT extension (editor+)>
```

```
<!ELEMENT editor EMPTY>
```

```
<!ATTLIST editor
```

```
  class          CDATA #REQUIRED
```

```
  expressionLanguage CDATA #REQUIRED
```

```
  label          CDATA #REQUIRED
```

```
>
```

The editor used for editing expression languages.

- **class** - The class which must implement org.eclipse.bpel.ui.expressions.IExpressionEditor or alternatively extend org.eclipse.bpel.ui.expressions.AbstractExpressionEditor.
- **expressionLanguage** - The URI of the expression language that identifies this expression editor.
- **label** - The user friendly label describing this expression language.

Examples: The following example illustrates the XPath expression editor which is provided within the org.eclipse.bpel.ui plugin.

```
<extension point="org.eclipse.bpel.ui.expressionEditors">
  <editor
    label="%XPATH_LABEL"
    class="org.eclipse.bpel.ui.expressions.XPathExpressionEditor"
    expressionLanguage="http://www.w3.org/TR/1999/REC-xpath-19991116">
  </editor>
</extension>
```

API Information: The value of the attribute class must represent a fully qualified name of a class which implements org.eclipse.bpel.ui.expressions.IExpressionEditor.

Alternatively can be a class which extend
org.eclipse.bpel.ui.expressions.AbstractExpressionEditor.

Supplied Implementation: The plugin org.eclipse.bpel.ui provides the XPath 1.0 expression editor. It also provides a default expression editor for expressions for which the expression language (identified by the namespace URI) does not match any registered expression editors.

In such cases, a very simple opaque editor is provided.

Actions

Identifier: org.eclipse.bpel.ui.actions

Since: [Enter the first release in which this extension point appears.]

Description: [Enter description of this extension point.]

Configuration Markup:

```
<!--ELEMENT extension (category+ , action+)-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id     CDATA #IMPLIED
  name   CDATA #IMPLIED
-->
```

```
<!--ELEMENT action EMPTY-->
<!--ATTLIST action
  id           CDATA #REQUIRED
  categoryId   CDATA #REQUIRED
  class        CDATA #REQUIRED
  specCompliant (true | false)
-->
```

- id -
- categoryId -
- class -
- specCompliant -

```
<!--ELEMENT category EMPTY-->
<!--ATTLIST category
  name CDATA #REQUIRED
  id   CDATA #REQUIRED
-->
```

- name -
- id -

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]

Model Listener

Identifier: org.eclipse.bpel.ui.modelListener

Since: [Enter the first release in which this extension point appears.]

Description: [Enter description of this extension point.]

Configuration Markup:

```
<!ELEMENT extension (listener)>
<!--ATTLIST extension
    point CDATA #REQUIRED
    id CDATA #IMPLIED
    name CDATA #IMPLIED
-->
```

```
<!--ELEMENT listener EMPTY-->
<!--ATTLIST listener
    class CDATA #IMPLIED
    id CDATA #REQUIRED
-->
```

- **class** -
- **id** -

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]

uiObjectFactories

Identifier: org.eclipse.bpel.ui.uiObjectFactories

Since: [Enter the first release in which this extension point appears.]

Description: [Enter description of this extension point.]

Configuration Markup:

```
<!--ELEMENT extension EMPTY-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id    CDATA #IMPLIED
  name  CDATA #IMPLIED
-->

<!--ELEMENT factory EMPTY-->
<!--ATTLIST factory
  id            CDATA #REQUIRED
  categoryId    CDATA #REQUIRED
  class         CDATA #REQUIRED
  specCompliant (true | false)
-->



- id -
- categoryId -
- class -
- specCompliant -

```

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]

Validator Factory

Identifier: org.eclipse.bpel.validator.org.eclipse.bpel.validator.factory

Since: 1.0.0

Description: Allow for addition validator factories to be registered with the BPEL validator.

Configuration Markup:

```
<!--ELEMENT extension (factory)+-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id    CDATA #IMPLIED
  name  CDATA #IMPLIED
-->
```

```
<!--ELEMENT factory EMPTY-->
<!--ATTLIST factory
  class CDATA #REQUIRED
  name  CDATA #IMPLIED
-->
```

- **class** -
- **name** -

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]

Model Query Interface

Identifier: org.eclipse.bpel.validator.modelQuery

Since: [Enter the first release in which this extension point appears.]

Description: [Enter description of this extension point.]

Configuration Markup:

```
<!--ELEMENT extension (modelQuery)-->
<!--ATTLIST extension
  point CDATA #REQUIRED
  id     CDATA #IMPLIED
  name   CDATA #IMPLIED
-->
```

```
<!--ELEMENT modelQuery EMPTY-->
<!--ATTLIST modelQuery
  class CDATA #REQUIRED
  name  CDATA #IMPLIED
-->
```

- **class** -
- **name** -

Examples: [Enter extension point usage example here.]

API Information: [Enter API information here.]

Supplied Implementation: [Enter information about supplied implementation of this extension point.]