How to create a new repository node?

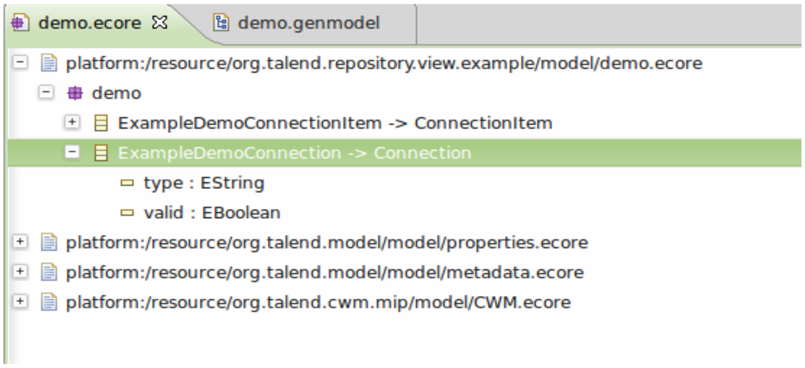
Now, I shall introduce the steps or how to create a new repository node. I think, it should be better for the OEM or other non-Talend's developers.

In order to understand more, I assume that:

- Based on **Talend v5.1.x**.
- Add a new type of repository node under “Metadata” in “DI” product.
- Enable create ExampleDemoConnectionItem in “metadata/example_demo” folder.
- Created in the example plugin “org.talend.repository.view.example”.

Steps:

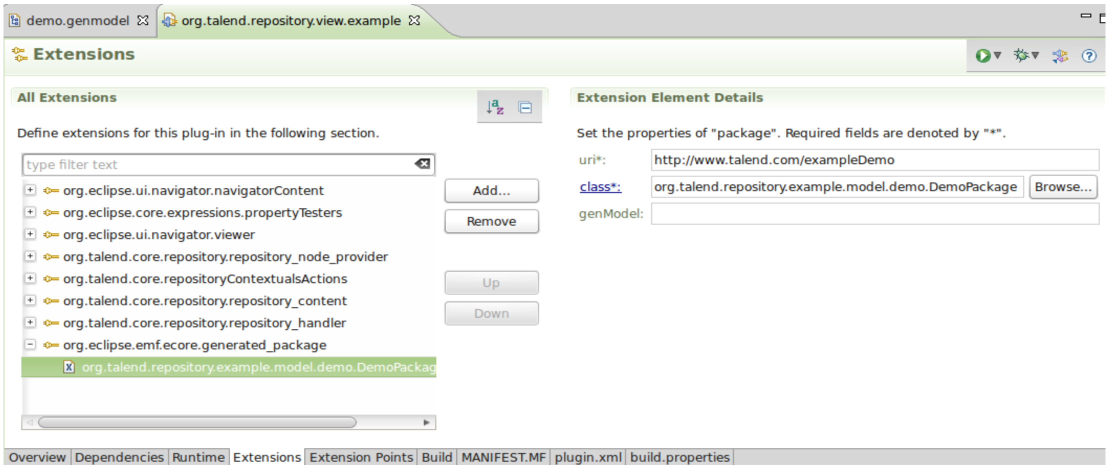
1) Create a EMF ecore model “demo.ecore”in the folder “model” of the example plugin.
   - ExampleDemoConnectionItem
   - ExampleDemoConnection with two attributes “type” and “valid”.



```
demo.ecore ⊠   demo.genmodel
⊟ platform:/resource/org.talend.repository.view.example/model/demo.ecore
  ⊟ demo
    ⊞ ExampleDemoConnectionItem -> ConnectionItem
    ⊟ ExampleDemoConnection -> Connection
        type : EString
        valid : EBoolean
⊞ platform:/resource/org.talend.model/model/properties.ecore
⊞ platform:/resource/org.talend.model/model/metadata.ecore
⊞ platform:/resource/org.talend.cwm.mip/model/CWM.ecore
```

Then create the “demo.genmodel”. And set the base package “org.talend.repository.model”. Also, it's better to set the runtime version to 2.6.

It's better to register the demo package by extension point “`org.eclipse.emf.ecore.generated_package`” explicitly.



Define extensions for this plug-in in the following section.

type filter text

```
⊞ org.eclipse.ui.navigator.navigatorContent
⊞ org.eclipse.core.expressions.propertyTesters
⊞ org.eclipse.ui.navigator.viewer
⊞ org.talend.core.repository.repository_node_provider
⊞ org.talend.core.repositoryContextualsActions
⊞ org.talend.core.repository.repository_content
⊞ org.talend.core.repository.repository_handler
⊟ org.eclipse.emf.ecore.generated_package
    org.talend.repository.example.model.demo.DemoPackag
```

Set the properties of "package". Required fields are denoted by "*".

uri*: http://www.talend.com/exampleDemo

class*: org.talend.repository.example.model.demo.DemoPackage

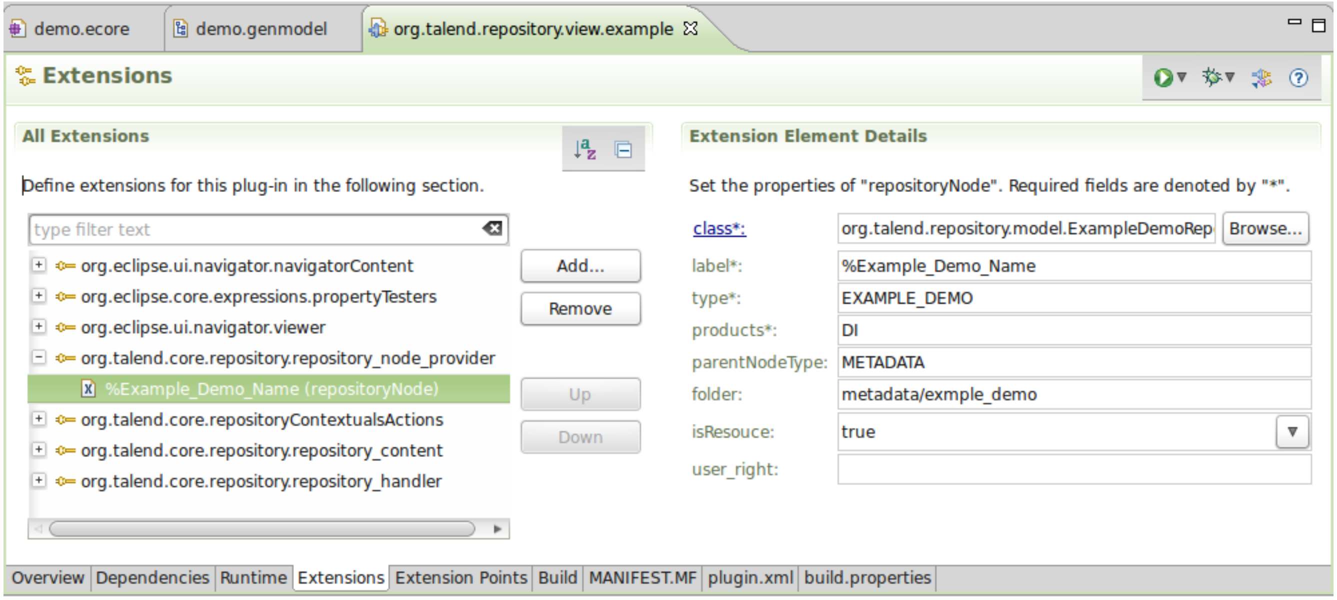
genModel:

Created by ggu, Aug 10, 2012

NOTE:

- Make sure add the dependencies for TOS plugin “org.talend.model” and “org.eclipse.emf.ecore.xmi”.
- In order to re-use the TOS models, need load some emf models from the plugin “org.talend.model” or “org.talend.cwm.mip”, like Properties, Metadata, CWM, TalendFile, Component, etc.

2) Add extension point “org.talend.core.repository.repository_node_provider” to define the root node for new repository node. Of course, need add the depended plugin “org.talend.core.repository” in the tab “Dependencies”.



All Extensions

Define extensions for this plug-in in the following section.

- org.eclipse.ui.navigator.navigatorContent
- org.eclipse.core.expressions.propertyTesters
- org.eclipse.ui.navigator.viewer
- org.talend.core.repository.repository_node_provider
  - %Example_Demo_Name (repositoryNode)
- org.talend.core.repositoryContextualsActions
- org.talend.core.repository.repository_content
- org.talend.core.repository.repository_handler

Extension Element Details

Set the properties of "repositoryNode". Required fields are denoted by "*".

class*: org.talend.repository.model.ExampleDemoRep
label*: %Example_Demo_Name
type*: EXAMPLE_DEMO
products*: DI
parentNodeType: METADATA
folder: metadata/exmple_demo
isResouce: true
user_right:

class: “ExampleDemoRepositoryNode”. Must implement the interface:
“org.talend.core.repository.IExtendRepositoryNode”

label: Will display in the repository view as root node.

type: will be used in the class “ExampleDemoRepositoryNodeType”(introduce it later).

products: set “DI” by default. Can be set like “CAMEL”, “DQ”. If support multi product, those value are split by “|”. For example, can set it like “DI|DQ”.

parentNodeType: here, because want to add the new node under the “Metadata”, so need set the “Metadata” type which is “METADATA”.

folder: it related to the attribute “isResource”, if set the “isResource” true, should provide a folder. Here, because it's under the “Metadata” and save all items in subfolder of metadata. So set the value like “metadata/exmaple_demo”.

isResource: If false, and no folder setting, it's only a type of repository node. There is no related EMF Item to be created.

user_right: Used in remote for User Authorization. If there is no right for the user, will hide this root node.

About the methods of class ExampleDemoRepositoryNode:

a. Method “*getNodeImage*”:

need create the “EexampleDemoImage”.

b. Method “*getOrdinal*”:

the order of node in repository view. (unused mostly)

c. Method “*getChildren*“:

add some children nodes by manually.
(normally, return empty array).

Additional:

> Create the enum “EExampleDemoImage” and implement the interface “IImage”.

The icons should be in the folder “/icons” of example plugin.

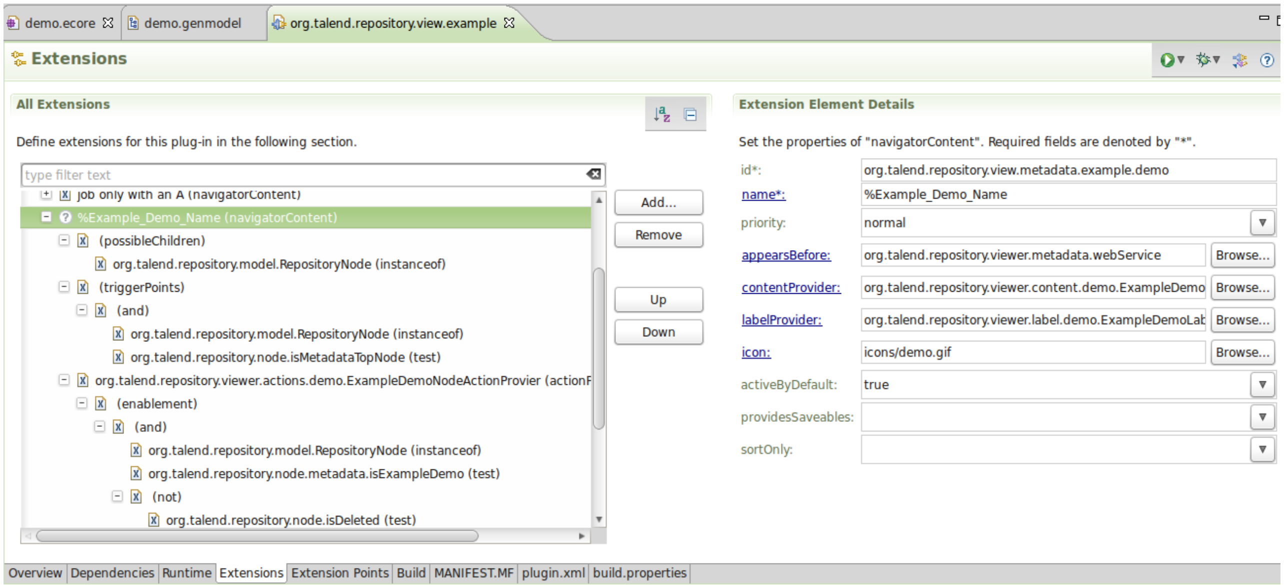
> Create the class “ExampleDemoRepositoryNodeType” to get the ERepositoryObjectType from the extension point by the type name(in my case, it's “EXAMPLE_DEMO”).

NOTE: need add the dependencies for plugins:

- org.talend.common.ui.runtime
- org.talend.commons.runtime
- org.talend.core.runtime
- org.talend.core.repository
- org.talend.repository.view
- org.talend.repository.metadata

Also see the plugin: “org.talend.camel.designer”.

3) Add the extension point “org.eclipse.ui.navigator.navigatorContent” for the demo node.



Define extensions for this plug-in in the following section.

```
+ Job only with an A (navigatorContent)
− %Example_Demo_Name (navigatorContent)
    − (possibleChildren)
        org.talend.repository.model.RepositoryNode (instanceof)
    − (triggerPoints)
        − (and)
            org.talend.repository.model.RepositoryNode (instanceof)
            org.talend.repository.node.isMetadataTopNode (test)
    − org.talend.repository.viewer.actions.demo.ExampleDemoNodeActionProvier (actionF
        − (enablement)
            − (and)
                org.talend.repository.model.RepositoryNode (instanceof)
                org.talend.repository.node.metadata.isExampleDemo (test)
                − (not)
                    org.talend.repository.node.isDeleted (test)
```

Set the properties of "navigatorContent". Required fields are denoted by "*".

id*:	org.talend.repository.view.metadata.example.demo
name*:	%Example_Demo_Name
priority:	normal
appearsBefore:	org.talend.repository.viewer.metadata.webService
contentProvider:	org.talend.repository.viewer.content.demo.ExampleDemo
labelProvider:	org.talend.repository.viewer.label.demo.ExampleDemoLat
icon:	icons/demo.gif
activeByDefault:	true
providesSaveables:	
sortOnly:	

Need implement the “ContentProvider” and “LabelProvider”. And config the “possibleChildren” and “triggerPoints”, “actionProvider” also.

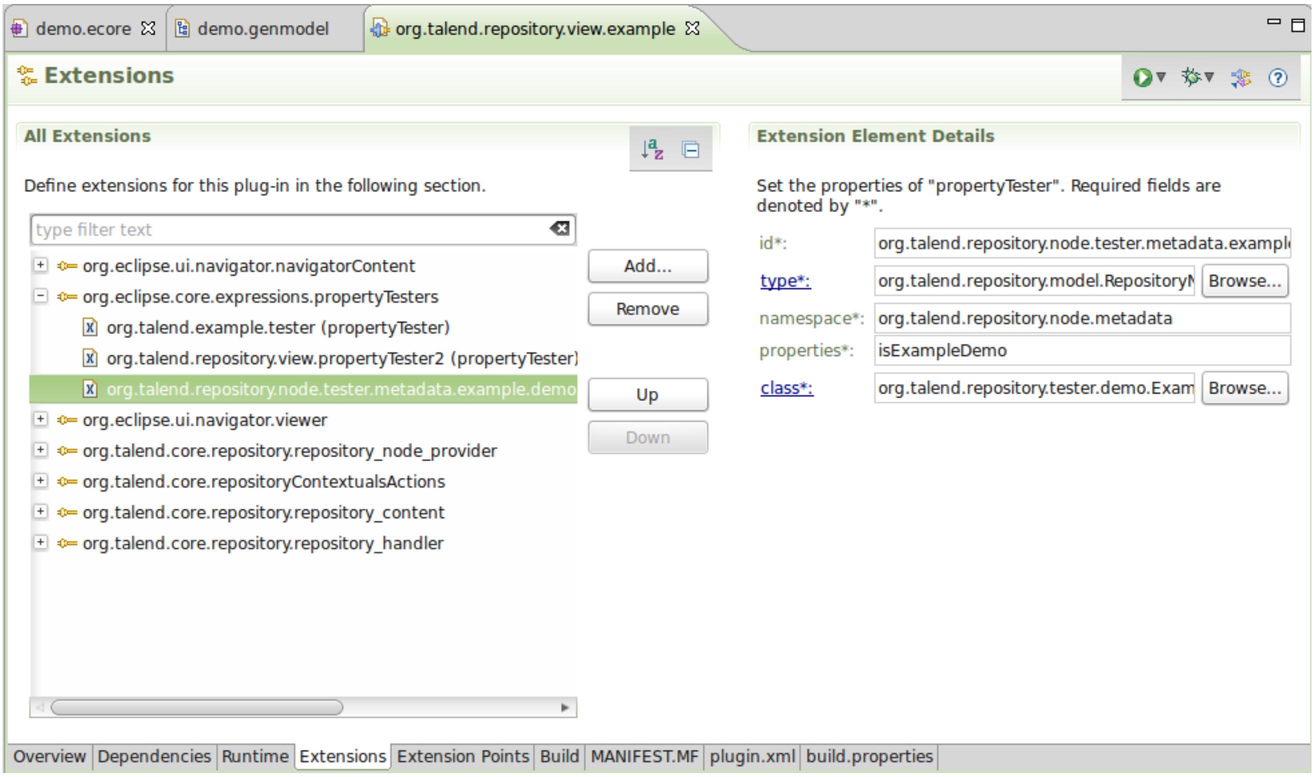
– **ContentProvider** (ExampleDemoContentProvider):
  If it's under the Metadata, extend from “AbstractMetadataContentProvider”.
  If it's like “Job Designer”, extend from “ProjectRepoDirectChildrenNodeContentProvider”.
  Will return the root node for the demo type in method “getTopLevelNodeFromProjectRepositoryNode”.

– **LabelProvider** (ExampleDemoLabelProvider):
  It extends from class “RepositoryViewLabelProvider”.
  If want do some specially. Do it in the two methods “getText” and “getImage”.

– **ActionProvider** (ExampleDemoNodeActionProvier):
  It extends from class “MetedataNodeActionProvier”.
  If general nodes, should extend from class “RepoNodeActionProvider”.

NOTE:
If want to i18n some strings in extension point, need add the following value in the “MANIFEST.MF”:
Bundle-Localization: plugin

4) In order to make sure the action provider to work well by “enablement”. Will add the extesion point “org.eclipse.core.expressions.propertyTesters”.



Extensions

All Extensions

Define extensions for this plug-in in the following section.

type filter text

- org.eclipse.ui.navigator.navigatorContent
- org.eclipse.core.expressions.propertyTesters
  - org.talend.example.tester (propertyTester)
  - org.talend.repository.view.propertyTester2 (propertyTester)
  - org.talend.repository.node.tester.metadata.example.demo
- org.eclipse.ui.navigator.viewer
- org.talend.core.repository.repository_node_provider
- org.talend.core.repositoryContextualsActions
- org.talend.core.repository.repository_content
- org.talend.core.repository.repository_handler

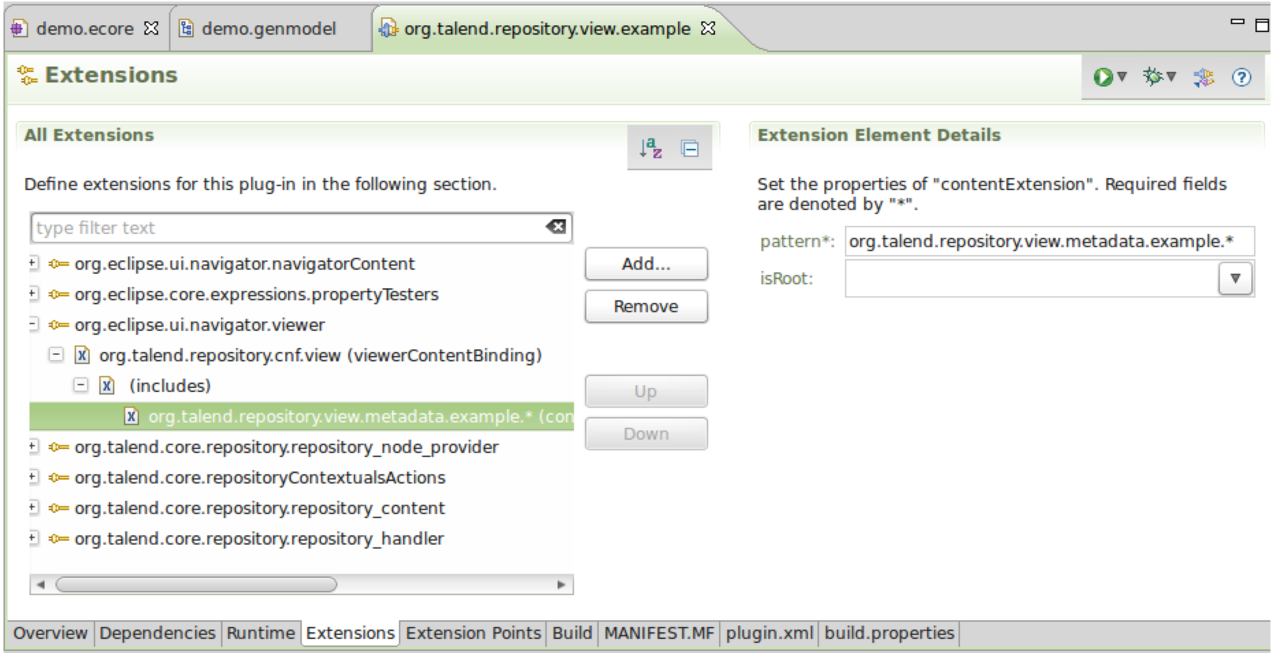
Extension Element Details

Set the properties of "propertyTester". Required fields are denoted by "*".

id*: org.talend.repository.node.tester.metadata.exampl
type*: org.talend.repository.model.RepositoryN
namespace*: org.talend.repository.node.metadata
properties*: isExampleDemo
class*: org.talend.repository.tester.demo.Exam

Need create a tester class “ExampleDemoMetadataNodeTester”.
About the value of attribute “properties” is “isExampleDemo” which must be same as the settings in the step 3) for the navigator content.

5) Also need add the extension point “org.eclipse.ui.navigator.viewer” for “viewerContentBinding” .


demo.ecore
demo.genmodel
org.talend.repository.view.example
Extensions
All Extensions
Define extensions for this plug-in in the following section.
type filter text
org.eclipse.ui.navigator.navigatorContent
org.eclipse.core.expressions.propertyTesters
org.eclipse.ui.navigator.viewer
org.talend.repository.cnf.view (viewerContentBinding)
(includes)
org.talend.repository.view.metadata.example.* (con
org.talend.core.repository.repository_node_provider
org.talend.core.repositoryContextualsActions
org.talend.core.repository.repository_content
org.talend.core.repository.repository_handler
Add...
Remove
Up
Down
Extension Element Details
Overview
Dependencies
Runtime
Extensions
Extension Points
Build
MANIFEST.MF
plugin.xml
build.properties

Set the properties of "contentExtension". Required fields are denoted by "*".

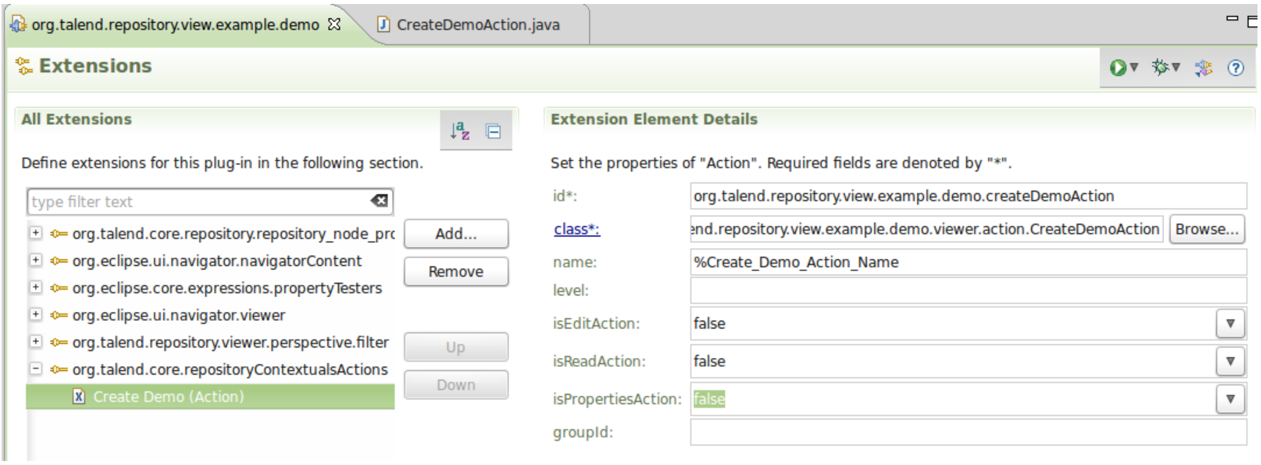
pattern*: org.talend.repository.view.metadata.example.*

isRoot:

Make sure the related CNF view enable the “navigatorContent” with pattern “org.talend.repository.view.metadata.example.*” the navigator content id in step 3) for the extension point “org.eclipse.ui.navigator.navigatorContent”.

6) If neeed and don't want this node to display in other perspective, for example, after switch to “Mediation” Perspective. Need add the extension point “org.talend.repository.viewer.perspective.filter”. And set the “includes” and “excludes”.

7) If want to add some actions on the repository node by right context menu. Need add the extension point “org.talend.core.repositoryContextualsActions” for your items.



Extensions

All Extensions

Define extensions for this plug-in in the following section.

type filter text

- org.talend.core.repository.repository_node_prc
- org.eclipse.ui.navigator.navigatorContent
- org.eclipse.core.expressions.propertyTesters
- org.eclipse.ui.navigator.viewer
- org.talend.repository.viewer.perspective.filter
- org.talend.core.repositoryContextualsActions
  - Create Demo (Action)

Extension Element Details

Set the properties of "Action". Required fields are denoted by "*".

id*:	org.talend.repository.view.example.demo.createDemoAction
class*:	end.repository.view.example.demo.viewer.action.CreateDemoAction
name:	%Create_Demo_Action_Name
level:	
isEditAction:	false
isReadAction:	false
isPropertiesAction:	false
groupId:	

Because this extension was defined in plugin “org.talend.core”, so make sure to add the dependencies for the plugin “org.talend.core”. And must implement the abstract class “AcontextualAction”

a. Method “*init*”:

Check that it's enable or visible for this action.

b. Method “*doRun*”:

For example. Open a wizard to set the details of demo item.

Also, if want to add some actions manually, can do it in the method “fillContextMenu” of atcion provider class “ExampleDemoNodeActionProvier”.

8) In order to create some ExampleDemoItem(s). Need create some classes, for example:

- “CreateDemoAction” which extends from “AbstractCreateAction”(Because it's created in metadata, so do the action is more like the “File Delimited”).

- “ExampleDemoWizard”, and some related “ExampleDemoWizardPage” or some step forms “ExampleDemoStep*Form”.

9) Then, must create a repository handler class “ExampleDemoRepositoryHandler” which implement the “IRepositoryContentHandler” to create/save item. And it's the extension point “org.talend.core.repository.repository_content”.

a. Method “*create*”:

Will be used to create the item in the factory.

b. Method “*save*”:

When save the item in the factory, will call this method.

c. Method “*createNewItem*”:

When duplicate a item by repository action “DuplicateAction”.

d. Method “*getRepositoryObjectType*”:

Return the type of item.

e. Method “*isRepObjType*”:

Check the type to be supported or not.

f. Methos “*getIcon*”:

The two methods will used by repository view and the export dialog to display the icon.

10) About the metadata connection for drag&drop. Need add the extension point “org.talend.core.repository.repository_handler” for the class “ExampleDemoDragAndDropHandler” which implement the interface “IDragAndDropServiceHandler”.

Also, if need can use interface “IRepositoryComponentDndFilter” with the extension point “org.talend.core.runtime.repositoryComponent_provider” for the dnd.

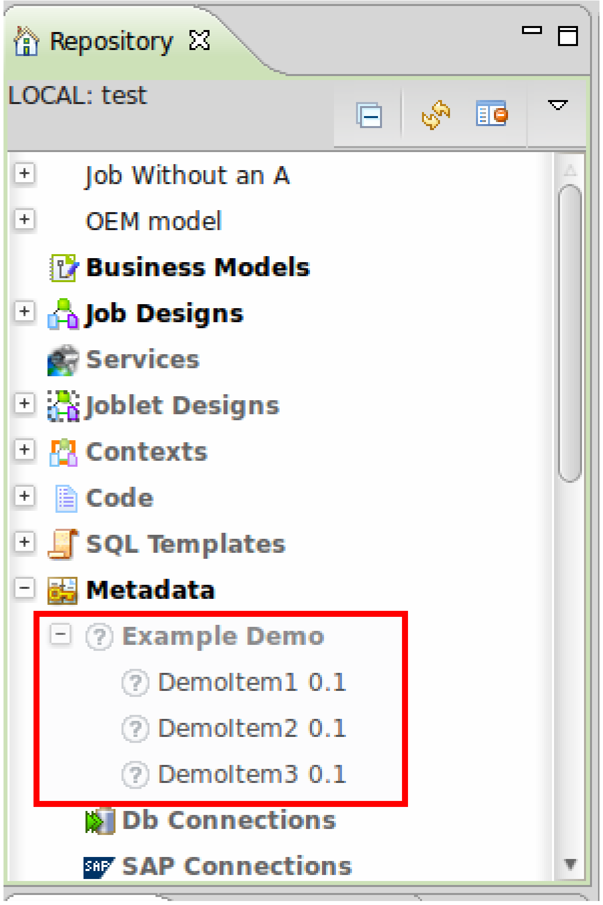
11) **Optional** :

- “Detect dependencies”: If want to do the update manager to work well , need check the class “RepositoryUpdateManager” and do some relationship in class “RelationshipItemBuilder” (in plugin:) also in plugin : org.talend.core.runtime.

- “Impact analysis”(Enterprise Edition): If want to do the analysis the related item. Need check the class “DataLineageManager” in the plugin org.talend.designer.datalineage.

BTW, Have tested the export and import function. They work well. But don‘t test the dnd and update manager, because there is no related components to test for this.

Demo Screenshot :



Job Without an A

OEM model

Business Models

Job Designs

Services

Joblet Designs

Contexts

Code

SQL Templates

Metadata

Example Demo

DemoItem1 0.1

DemoItem2 0.1

DemoItem3 0.1

Db Connections

SAP Connections