

Property Rendering

Exercises...

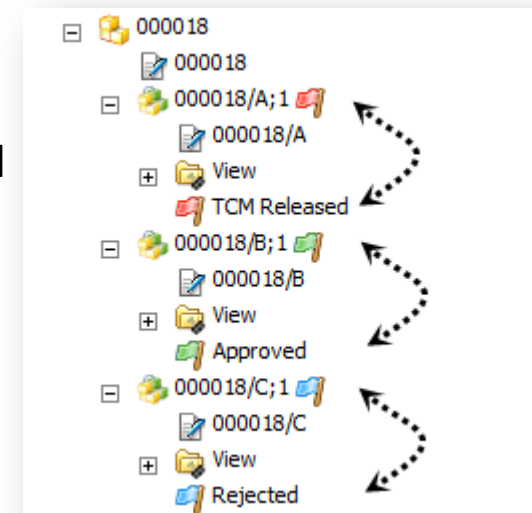
Change an existing business object icon

See *BMIDE guide page 5-45*

Maybe there are better, easier ways to implement the following examples, but... these work!

Exercise 1:

- Having the same icon for *different Release Statuses* behind the released object (e.g. ItemRevision) and the Releases Status object itself.
- For the Release Status object *Reject, Approved* and *TCM Released* a **Blue**, **Green** and **Red** flag will be defined.



Steps to perform

1) Create the icons

As default icon for a Release Status a flag pole will be defined.

For Release Status “*Reject*” a **blue** overlay flag will be defined

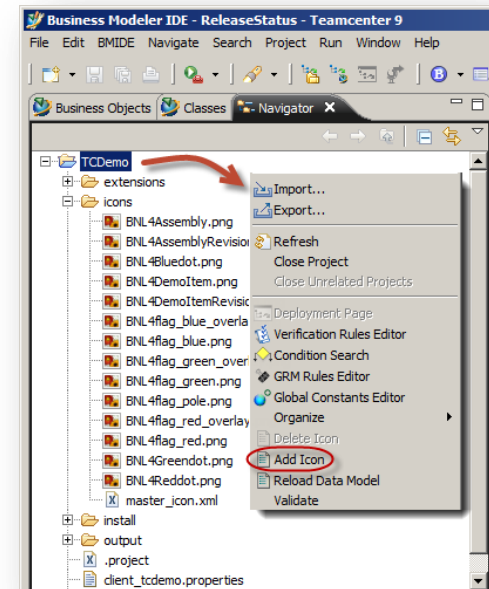
For Release Status “*Approved*” a **green** overlay flag will be defined

For Release Status “*TCM Released*” a **red** overlay flag will be defined



2) Add icons to the BMIDE project

- In the Navigator view, right-click your project and choose Add Icon.
- Add the icons you want to import...
- Notice that the icons will receive the project prefix

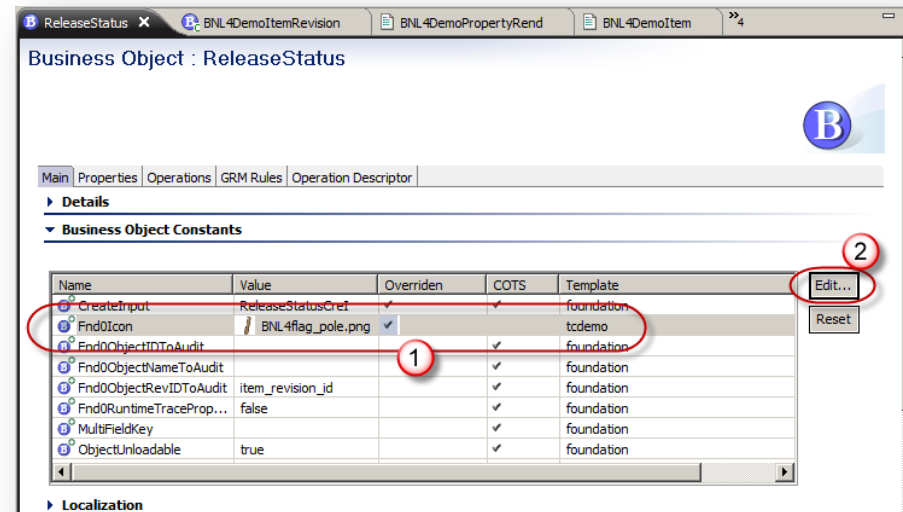


Steps to perform

3) Add icons to Business Object

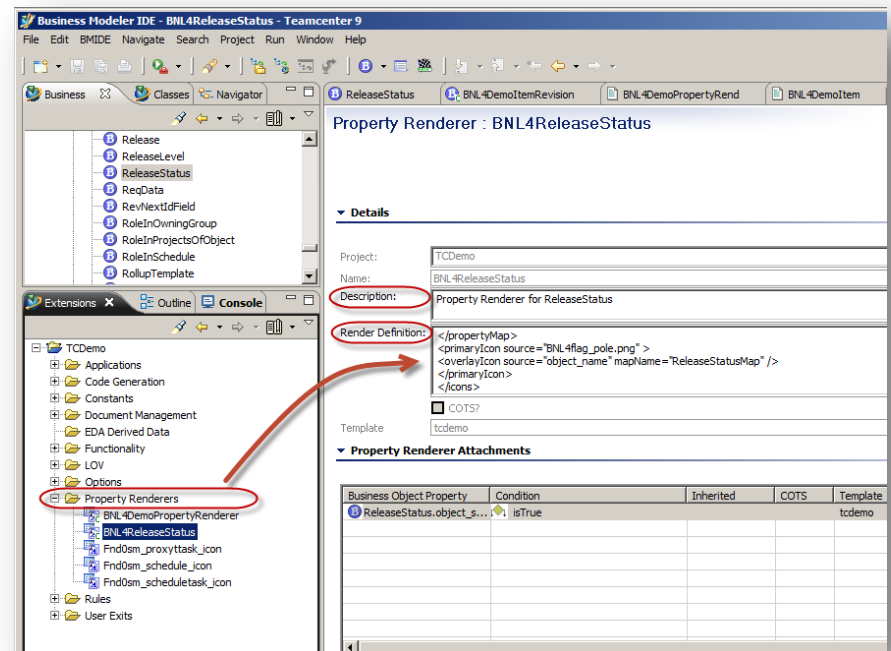
- Search for Business Object Release Status.
- Add the new “default” icon to the Release Status object by...
 - Select Fnd0Icon (Business Object Constants) and select the Edit... button
 - Select the icon from the (project) list using the Browse button

Now, the result would be that default “checkered flag” is replaced by the “flag pole”



Steps to perform

- 4) Create a property renderer to associate icons with properties
 - In the Extensions view, right-click the Property Renderers folder and choose New Property Renderer.
 - Type in a Name and a Description.
 - Create a Render Definition (see next slide)



Steps to perform

Render definition XML format (see BMIDE guide 5-53/54)

- Icons
- primaryIcon
- overlayIcon
- propertyMap

Overlay icon (source=object_name) will be determined by the ReleaseStatus.object_name and it matching value from the ReleaseStatusMap (“Rejected”, “Approved” and “TCM Released”)

```
<?xml version="1.0" encoding="UTF-8"?>
<icons Version="1.0">
  <propertyMap name="ReleaseStatusMap">
    <item key="Rejected" value="BNL4flag_blue_overlay.png"/>
    <item key="Approved" value="BNL4flag_green_overlay.png"/>
    <item key="TCM Released" value="BNL4flag_red_overlay.png" />
  </propertyMap>
  <primaryIcon source="BNL4flag_pole.png" >
    <overlayIcon source="object_name" mapName="ReleaseStatusMap" />
  </primaryIcon>
</icons>
```

Steps to perform

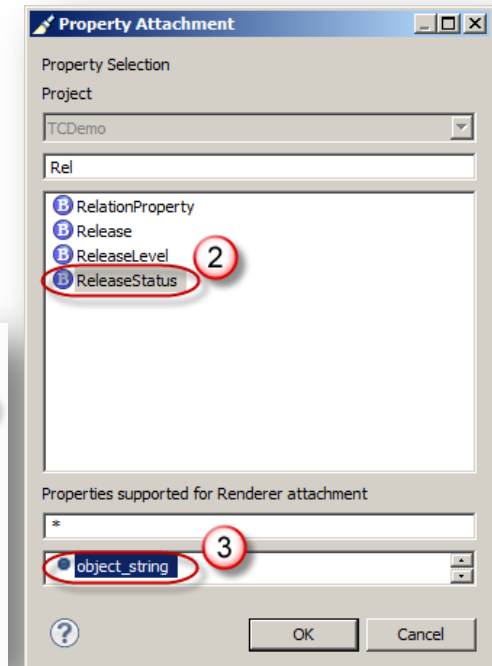
5) Attach the property renderer to the business object.

- Select the *Attach* button
- Select *ReleaseStatus*
- Select *object_string*

Note For any subtypes of *WorkspaceObject* types, only renderers that are attached to *object_string* property are effective. see *BMIDE guide 5-50*

▼ Property Renderer Attachments

Business Object Property	Condition	Inherited	COTS	Template	
ReleaseStatus.object_string	isTrue			tcdemo	Attach Detach



Steps to perform

6) Deploy your project by...

- Saving the Data model (BMIDE→ Save Data Model)
- Create a package template using the Package Template Extensions wizard and deploy using the TEM.
- At this point, the result would be that default “checkered flag” icon of the Release Status object is replaced by the “flag pole” with matching overlay icon.

Steps to perform

- 7) To change the status flag icon of the object that has been released (e.g. the icon behind an ItemRevision) the 'old' method is still valid.
- Create a `customer.properties` file in the `<TC install>\portal\plugins\configuration_9000.1.0` directory and add contents for custom Release Status icons
 - Create an `images` sub-directory in this directory and place the icon files
 - Start `<TC install>\portal\registry\genregxml.bat`
 - Contents of the `customer.properties` file:

```
# Custom type icons
# -----
release_status_list.TCM\ Released.ICON=images/flag_red.png
release_status_list.Rejected.ICON=images/flag_blue.png
release_status_list.Approved.ICON=images/flag_green.png
```
 - At this point, the result is as defined...

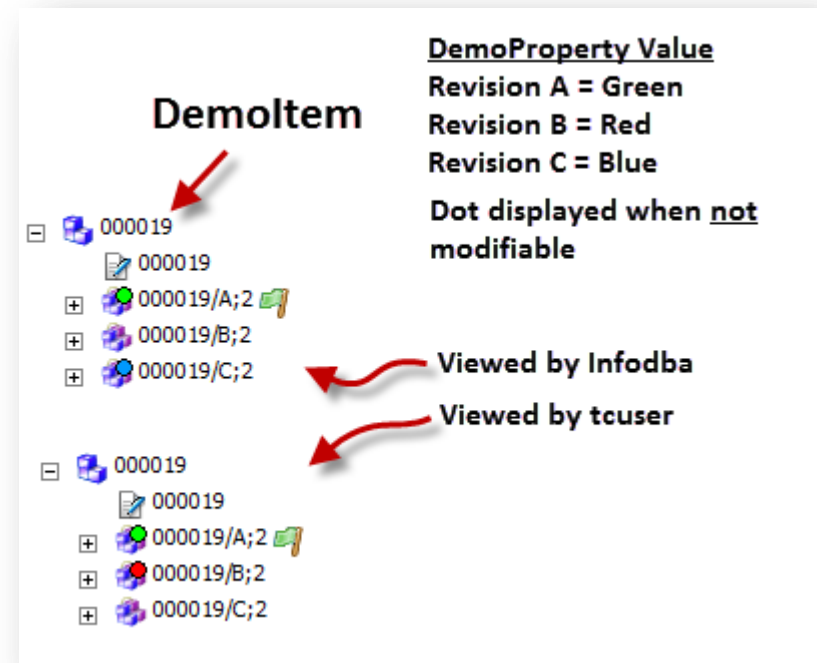
Add a custom business object icon

Exercise 2:

- ❑ Defining a custom business object with icon and different icon overlays, depending on the value of a custom property visible only depending on the value of a second property.

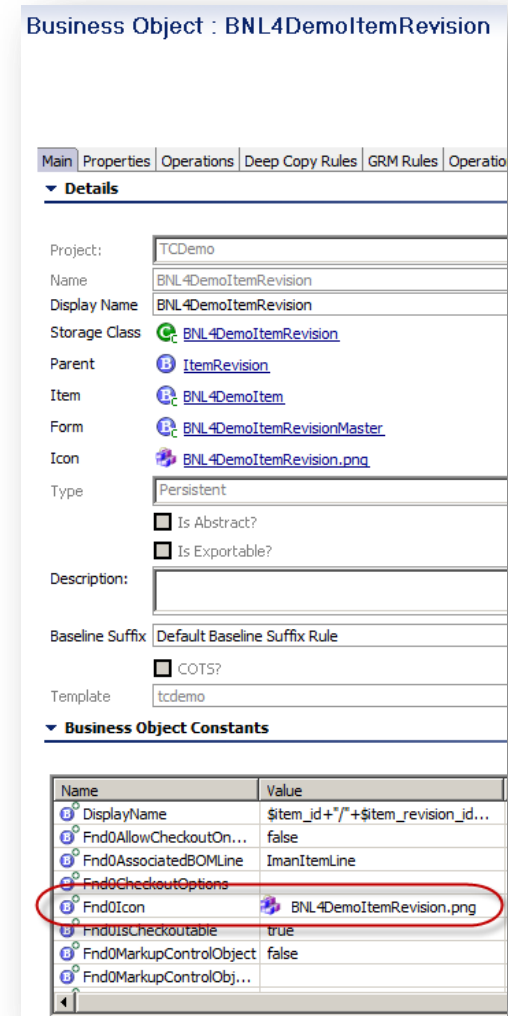
- ❑ In detail:
A custom Item Type “*Demo Item*” having an ItemRevision property “*DemoProperty*” with values **Blue**, **Green** or **Red**.

- ❑ A **Blue**, **Green** or **Red** dot will be displayed over the “*DemoItemRevision*” icon depending on the value of the “*DemoProperty*” *only when the “*DemoItemRevision*” is not modifiable* (*is_modifiable=false*).



Steps to perform

- 1) Create and add new icons to the BMIDE project (for “Demo Item”, “Demo ItemRevision” and overlay icons, “Blue dot”, “Green dot” and “Red dot”).
- 2) In the BMIDE create a new Item Type: “Demo Item” (e.g. BNL4DemoItem)
 - Attach the new icon for “Demo Item”
 - Attach the new icon for “Demo ItemRevision”
- 3) Add a “DemoProperty” to the “Demo ItemRevision” (e.g. bnl4DemoProperty)
- 4) Create and attach a “DemoLoV” with values **Blue**, **Green** and **Red** to the “DemoProperty”
- 5) At this point (after deploy), the new “Demo Item”, “Demo ItemRevision” with DemoProperty and DemoLoV would be available.



Steps to perform

6) Create a new "Property Renderer" (e.g. BNL4DemoPropertyRenderer)

```
<?xml version="1.0" encoding="UTF-8"?>
<icons Version="1.0">
  <propertyMap name="DemoMap">
    <item key="Blue" value="BNL4Bluedot.png"/>
    <item key="Green" value="BNL4Greendot.png"/>
    <item key="Red" value="BNL4Reddot.png"/>
  </propertyMap>
  <primaryIcon source="BNL4DemoItemRevision.png"/>
  <overlayIcon source="bnl4DemoProperty" mapName="DemoMap">
    <visibleWhen>
      <with variable="is_modifiable">
        <not>
          <equals value="Y"/>
        </not>
      </with>
    </visibleWhen>
  </overlayIcon>
</icons>
```

Business Object **BNL4DemoItemRevision**

Main Properties Operations Deep Copy Rules GRM Rules Operation Desc

Property Name	Type	Storage Type
object_desc	Attribute	String[240]
object_name	Attribute	String[128]
object_properties	Attribute	Short
object_string	Runtime	String[32]
object_type	Attribute	String[32]
owning_group	Reference	TypedReference
owning_organization	Reference	TypedReference
owning_project	Reference	TypedReference

Property Constants

Naming Rule Attaches of object_string

LOV Attaches of object_string

Property Renderer Attaches of

Property Renderer	Condition
BNL4DemoPropertyRenderer	isTrue

7) Attach the newly created "Property Renderer" to BNL4DemoItemRevision.object_string

Steps to perform

8) Deploy your project by...

- Saving the Data model (BMIDE→ Save Data Model)
- Create a package template using the Package Template Extensions wizard and deploy using the TEM.
- At this point, the result will be that a **Blue**, **Green** or **Red** dot will be displayed over the “*DemoltemRevision*” icon depending on the value of the “*DemoProperty*” *only when the “DemoltemRevision” is not modifiable.*