You can customize or add entirely new work items in IBM® Rational Team Concert™ project management tools to fit your team's needs. The team in this example uses agile development methods and the scrum process. Their recent experience shows quite a few regressions, which indicates that they need to ensure that any defect fix is thoroughly tested. Therefore, they need to associate an attribute with the Defect type of work item, and the value of this attribute should be based on team members' roles to make sure that only testing team members can be assigned. This article explains how to create a new role-based value set and associate it with a Defect type of custom work item created in Part 1, using the Rational Team Concert Eclipse client rather than the web client.

Before you start

To follow steps in this article, you need to have following software running and configured properly:

- IBM® Rational Team Concert™ version 3.0, which runs on the IBM® Rational® Jazz™ server
- Rational Team Concert 3.0, client for the Eclipse IDE

This software can be installed and configured according to instructions provided in this previous developerWorks article: "Tutorial: Create a custom work item in Rational Team Concert" (see Related topics for a link). After completing that tutorial, you will have a local IBM Rational Jazz server running, named localrtcjazz.com. It can be accessed using this public URI: https://localrtcjazz.com:9443/ccm. The default administrator access on this server has been disabled, and a new admin user has been added, called seemag, because these customizations must be done by an administrator-level user.

1. After you have the server running, open the Rational Team Concert client for the Eclipse IDE, and establish a connection to repository created as above by using the option to Create a new jazz repository connection.
2. Provide the public URI, https://localrtcjazz.com:9443/ccm, and the admin user name and password for the repository.
Use case scenario for this article

You are part of a project team that has been maintaining an earlier version of a product, and they have observed a lot of regressions in the defect fixes that have been provided. So the project manager decides that, for each and every defect, a tester needs to be identified, and this should be tracked for accountability. The major requirement is that the tester identified for each defect should belong to the testing team of the project. Rational Team Concert can be used to address this requirement by creating a custom role in the project called "Tester" and then creating value sets based on that role. Then the value sets can be associated with any work item and used as-is.

Before you can customize work items, you need to have a project area named TestProject created. For this article, it has been created already by using the Scrum Process template.

Customize a Defect type of work item, based on a role

The requirement of the project team discussed in the use case scenario can be addressed if you associate each defect work item with a custom attribute that can hold values from a value set. A value set is a set of values of project team members who are assigned to a Tester role.

The following sections describe details for these steps to address this requirement, using Rational Team Concert Eclipse client:

- Create a custom role in the project called Tester, and assign it to a few team members.
- Create a value set based on the Tester role.
- Create a custom attribute called Tester for the Defect type of work item that is associated with the new value set.
- Modify Defect editor presentations to show that custom attribute.
- Bind the modified presentation with the Defect type of work item.

Create a Tester role in the project

There are three steps to create a new role in the project as:

1. Add a new role.
2. Assign permissions to the new role.
3. Associate this role with a few team members of the project.

Add a new role

1. To add a new role, select Process Configuration > Roles (Figure 1), and click the Add Role button (to the right of the Defined Roles label: a yellow figure symbolizing the user, with a plus addition sign).
2. For the Name field, enter Tester, and click Apply changes.
Assign permissions

By default, Rational Team Concert does not provide any permissions to a newly created custom role. You must assign permissions as decided by project team.

1. To assign permissions, open the project in the Team Organization view, and click Process Configuration > Project Configuration > Permission. Select Tester as the role.

For this example, grant these permissions to the Tester role:

2. Create or modify a Defect type of work item:
   a. Scroll to the Save Work Item (server) section under Permitted actions, expand Create a work item.
   b. Expand Create a work item of specific type and select Create a 'Defect' work item.
   c. Also scroll down to Modify the work item and check box Modify the work item (Figure 2).

3. Save a query:
   a. From the Permitted Actions list, scroll to Work Items.
   b. Below the Work Items action, check the Save query (server) action (Figure 3). This will automatically select all of the actions under Modify a query.

Figure 2. Assigning permissions to a custom role: Modify the work item
Associate the role with a project

Now this new role needs to be assigned to testing team members of the project.

1. Ensure that the project is still opened in the Team Organization view and that you are in Overview tab. Any roles for a team member can be added or removed in this view.
2. To add or remove a role for a team member, select or highlight the name of that member. For this example, assign the Tester role to only two users: Jerry Thomas and Koyal.

As soon as the member name is selected, notice that the Process Roles button shown in Figure 4 is automatically enabled.

Figure 4 below shows the team members after the Tester role has been assigned to Jerry Thomas and Koyal. Figure 5 shows the screen captured while the Tester role was being assigned to Jerry.

**Figure 4. Project Members in various roles assigned to them**
3. After it is enabled, click the Process Roles button to open a dialog window to add or remove roles, as shown in Figure 5.
4. Make sure that they have only Tester and the default role in the Assigned Roles on the right side.

**Figure 5. Add Tester role for designated users**

![Edit Process Roles](image)

Create a role-based value set

1. To create a role-based value set, select **Process Configuration > Project Configuration > Configuration Data > Work Items > Attribute Customizations > Value Sets**, and click **Add**.
2. This will open a dialog window to add a new configuration (Figure 6). Enter these values:
   - **Name**: Testers
   - **Category**: Value Sets
   - **Type**: Role-based user list
3. Click **OK**.

**Figure 6. Creating a role-based value set**

![Add Configuration](image)

The new value set named Testers will be created as shown in Figure 7.
This value set needs to be associated with a role from a project area.

4. Highlight the **Testers** value set and click the **Add Role** button at the right to open the dialog window in Figure 8.

5. In the Role field, select **Tester** from the drop-down menu, and click **OK**.

**Associate the new value set with the Tester attribute in the Defect work item**

1. To create a custom attribute for an existing Defect work item, select Process **Configuration > Project Configuration > Project Configuration Data > WorkItems > Types and Attributes**.

   Selecting the **Types and Attributes** tab will provide a list of all of the types of work items that are available with a standard scrum template, one of which is called Defect.

   2. Select **Defect** as shown in Figure 9.
3. Scroll down to the Attributes panel of the Defect work item (see Figure 10).

4. Make sure that the check box for **Show only custom attributes** is checked, so you see only custom-created attributes.

5. Click **Add** toward the right side of the Attributes panel to open the Add Custom Attribute dialog window shown in Figure 11.

6. For this example, enter this information:
   - **Name**: Tester
   - **ID**: tester
   - **Type**: Contributor
   - **Value Set**: Testers
Figure 11. Creating a Contributor type of custom attribute for a Defect work item

Modify defect editor presentations to show the custom attribute

An editor presentation is a way to present a work item to users. In current versions of Rational Team Concert, a work item can be displayed four different ways:

- **Work Item Editor**, the presentation used in the Eclipse client and the web client when displaying a work item in the Editor view
- **Inline Work Item Editor**, used in web client to display work item details in a Plan view
- **Lightweight Work Item Creation dialog**, used in the web client when creating work items
- **Plan Editor Preview**, displayed as a pop-up window in the Eclipse client when the user moves the cursor over a work item in the Plan Editor view

**Important:**
A work item can be viewed in any type of presentation; therefore, all four editor presentations need to be modified for any work item type. All of the editor presentations are created in same way. This article covers only modification of a Work Item Editor presentation for a defect, but you need to modify other three editor presentations too before using the modified version of Defect work item in the project area.
Rational Team Concert provides a default editor presentation that includes most of the fields, tabs, and so forth that are required to present any work item. The default layout can be changed by adding a few custom attributes.

1. To start adding or editing the Defect work item editor presentation, go back to Process Configuration > Project Configuration > Project Configuration Data > WorkItems > Editor Presentations.

2. Observe the text box titled Choose the editor presentation to edit. Retain the default value of com.ibm.team.workitem.editor.default, and click Duplicate (highlighted in Figure 12).

Figure 12. Default editor presentations in Rational Team Concert

3. A dialog window will open for you to provide the ID for the duplicated editor presentation. Give it a suitable name, such as DefectWorkitemPresentation, and click OK.

The new editor presentation will be added (see Figure 13).

After a new presentation is created for a Defect work item (DefectWorkitemPresentation), it needs to be modified to add a new attribute. A presentation or view is divided into various tabs, sections, and presentations. The Header, Overview, Links, Approvals, and so on form the tabs, and each tab has more parts. For example, the Overview tab contains all details of a work item, and links refer to links for a work item. Depending on where you want to change the view, the editor presentation needs to be modified. For this example, an attribute needs to be added in Details of a work item that can hold the value of Tester. Details form part of Overview tab, so you need to start modifications there.
Figure 13. DefectWorkItemPresentation created, based on default

![DefectWorkItemPresentation created, based on default](image)

**Important:**
Do not modify any shared content directly in an editor presentation, because it would be shared among multiple editor presentations.

The Overview tab in Defect Work Item Editor presentation has a "content shared" notation associated with it, and this should be duplicated before making any modifications.

1. Select **Overview**, and click **Duplicate**.
2. In the dialog window that opens, provide an appropriate name (**Overview** in this case), and click **OK**. This will create an additional Overview tab that will not have a [content shared] notation. After the Overview tab is created, it's time to modify the Details sections to show Tester custom attribute in the Work Item Editor presentation.

Again, the Details section is shown with a [content shared] notation; therefore, it needs to be duplicated first.

3. Select **Details**, and click **Duplicate** again.
4. When you are prompted for an ID for duplicated section, provide an appropriate name and click **OK**.

At this stage, there will be two Overview tabs and two Details sections, one with the [content shared] notation and one without it, as Figure 14 shows.
Figure 14. DefectWorkitemPresentation after adding Overview and Details

Because you have Overview tabs and Details sections created specifically for the Defect work item presentation, you don't need the shared Overview tab and Details section anymore.

5. To remove those sections or particular elements of the editor presentation, select the appropriate element (Overview [shared content] and Details [shared content]), and click Remove.

After you remove the shared Overview tab and Details, notice that you have only one Overview and Details section, without any shared content in the DefectWorkitemPresentation view. Now you need a presentation for the custom attribute that you want to show in the Defect work item editor: Tester.

6. Expand the Details section by clicking the plus sign + of the Overview tab, and then click Add Presentation to open a dialog window where you can provide values for presentation field.

7. The presentation can be attribute-based or not attribute-based. Because you are adding the presentation field for a custom attribute of a Defect work item, select Attribute-based presentation.
8. Complete the required fields:

- **Attribute**: Tester has been chosen as the attribute, because the presentation is being added for the Tester attribute of the Defect work item type.
- **Kind**: Select **Generic Combo**, which is a special kind of presentation provided by Rational Team Concert 3.0. It shows values directly from the value set associated with the attribute, without applying any additional logic.
- **Label**: A label is required for visual identification of the presentation field.

9. Click **OK**.

As Figure 16 shows, there is now a Tester presentation in the Details section.

**Figure 16. DefectWorkitemPresentation with additional presentation for Tester**

You can change the position of this presentation by clicking the **Move Up** or **Move Down** buttons.
You have finished creating a new Work Item Editor for the Defect work item.

**Reminder:**
Remember to add the other three types of Editor presentations to be able to view changes properly in all views.

**Bind the modified presentation with the Defect work item**

Although there is a new editor presentation created for Defect work item, the Defect is still associated with default editor presentations for all of its four editors, which should updated to use the newly created editor presentations that are specific to the Defect work item.

1. To update the editor presentation for Defect work item, select **Process Configuration > Project Configuration > Configuration Data > WorkItems > Types and Attributes**, and click **Defect**.
2. Scroll down to Work Item Editor attribute and For the **Work Item Editor** field, and select **DefectWorkitemPresentation** from the drop-down menu, as Figure 17 shows.

![Figure 17. Various editor attributes for Defect work items](image)

**Reminder:**
Remember to choose appropriate values for other three types of editor presentations, using the names that you gave them while creating new presentations.

**Test the role-based customization**

To test role-based customizations you added, create a new Defect (also see Figure 16):

1. Open the project in Team Artifacts view.
2. Select the project on left side, and select **Work Items**.
3. Right-click on **Work Item**, click **New**, and in the submenu, select **Defect**.

This will open a new defect, as shown in Figure 18.
As Figure 18 shows, the defect has an additional field: **Tester**. If you click the drop-down menu for Tester, the menu will list only those members from the project who have been assigned the role of a Tester in the project: Koyal and Jerry Thomas.

**Summary**

Rational Team Concert provides extensive process configuration options to meet requirements of almost every project. This article covered one of the customizations for work items. As a part of this customization, a new type of role and role-based value set were created and associated with a Defect type of work item. While testing, the attribute picks up values only from a value set that has been created based on a specific project role called Tester.