Rational Team Concert essentials: A developer's perspective, Part 2: Delivering work contributions

Ali Manji

July 28, 2015

Learn how to use Rational Team Concert to identify your project development commitments, carry out your development activities in a team environment, and deliver your work to your team project.

View more content in this series

Introduction

In Rational Team Concert essentials: A developer's perspective, Part 1: Joining a new team project, you learn about the main concepts behind Rational Team Concert's change management mechanisms. You also make use of Rational Team Concert and the Money That Matters Lifecycle sample application to join a team project, create a repository workspace, and load project components and artifacts.

In Part 2 of this article series, you continue to use Rational Team Concert V.5.0.2 to:

• Identify your work assignments
• Complete work-related development activities and tasks, and finally
• Deliver your work contributions back into your team repository

The goal of this article series is to serve as a helpful collection of "developer's cheat sheets." Some of the operations outlined are ones you might use infrequently. While others, you'll use often enough for them to become second nature to your normal development routines.

Identify work assignments

Before you start

The steps in this article continue to make use of the Rational Team Concert deployment and the user bob from the Money that Matters sample in Part 1. Completing the steps in Part 1 completely before proceeding with this part is essential.

Learn more. Develop more. Connect more.

The new developerWorks Premium membership program provides an all-access pass to powerful development tools and resources, including 500 top technical titles (dozens specifically for web developers) through Safari Books Online, deep discounts on premier developer events, video replays of recent O'Reilly conferences, and more. Sign up today.
Now that you have successfully joined the Money that Matters project team, you'll want to understand your work assignments.

In Rational Team Concert, work assignments come in the form of units of work known as a **Work Item**. Depending on your team's choice of available development process templates and any process customizations incorporated, you'll notice that there a number of Work Item types available to use for different types of work.

The Money that Matters agile project team uses the **scrum process**. With the scrum process, the Work Item types available are **Story, Defect, Task, Epic, Retrospective, Impediment, Adoption Item**, and **Track build item**. The Knowledge Center contains a great definition of the different **Work Item types**.

To identify bob's initial work assignments immediately after joining the project, follow these steps:

1. On your developer workstation, open Rational Team Concert/Eclipse with the same workspace used in Part 1.
2. Open a command prompt window and type `<eclipse_install_dir>\eclipse –data <workspaceName>` e.g. C:\eclipse\eclipse C:\myWorkspaces\MoneyThatMatters
   - If prompted for a login:
     - **User ID:** bob
     - **Password:** bob
3. Open the **Work Items** perspective, **Windows > Open Perspective > Work Items**.
4. Select the **My Work** view and click the **accept all work** link as shown in Figure 1.

**Figure 1. Accept your work upon joining a project**
5. The **Work Items** assigned to Bob now show up under the **Current Work** and **Future Work** sections of the **My Work** view listed in order by due date as shown in Figure 2.

**Figure 2. See your Work Items in the My Work view**

Within the **Work Items** perspective there is a view, also named **Work Items**. In this view you can also identify your current work assignments. To do this:

1. In the **Work Items** view, click the link **Open assigned to me (JKE Banking (Change Management))**.
2. You can see the **Id**, **Status**, **Priority**, **Severity**, a **Summary**, and who the Work Item was **Created By** for each Work Item. Figure 3 lists the Work Items assigned to bob.

**Figure 3. Your Work Items in the Work Items view**

3. In the **Team Artifacts** view, click the **Create a Repository Connection** link.
4. In the Create a Jazz Repository Connection dialog, type:
   a. `https://<your_JTS_server_hostname>:<jts_port>/ccm`
   b. Enter `bob` for both the **User ID** and **Password** fields and click **Finish**.

**Time to do some work!**

In addition to its strong change management capabilities, Rational Team Concert provides developers with tool and process guidance. This allows you to focus on what is most important: doing work.

Doing work for a developer usually entails creating new, or updating existing software artifacts, and quality tasks against new or updated artifacts. A software artifact can take the form of a new Java source class, a web page, a JavaScript file, a new Eclipse folder or even a new Eclipse project to house other source artifacts, and so on.
The work that you will carry out in the steps below will lead you to implement Work Item #4, as shown in Figure 4, with the description: Provide faceted search capabilities.

Share a new Eclipse Java Project with a new Rational Team Concert Source Component

1. Switch to the Java perspective.
2. Create a new Java project, by selecting File > New > Java Project.
3. In the Create a Java Project page of the New Java Project wizard, in the Project Name: field type JKESearchLogic and click Finish.
4. In the Package Explorer view, select the newly created JKESearchLogic project right-mouse click and select Team > Share Project....
5. On the Share Project page of the Share Project wizard, select Jazz Source Control. Click Next.
6. On the Select Component page, select the BRM Stream Workspace stream and click the New Component... button.
7. Enter Search Logic for the new component name in the New Component dialog box and click OK.
8. You should now see the Search Logic component in the list of components on the Select Component page of the Share Project in Jazz wizard, as shown in Figure 4. Click Next >.

Figure 4. Create a new Rational Team Concert Project Component for a new Eclipse project

9. Ensure that there is a check mark next to the JKESearchLogic Java project under the Available unshared projects list of the Projects page and click Next >.
10. Click **Finish** on the **Review Ignored Resources** page.

The **Pending Changes** view now shows up in the perspective as shown in Figure 5. This view identifies any changes you made in the course of doing your work that are awaiting check-in or delivery. Check-in involves associating changes made to a particular change-set (and ultimately to a Work Item). Delivering a change set (associated to a Work Item) makes the changes available to other members of your team so that they can build upon your work or make use of it in the course of their own development and test work efforts.

**Figure 5. Creating a new Project Component triggers the creation of a new change set**

Identifying your changes while doing work

Because the changes made so far were to create a new empty Java project and to associate it directly to a new Rational Team Concert component, Rational Team Concert went ahead, as a convenience, to create a change set for you with the new Java project already checked in.

Figure 5 shows the new Rational Team Concert component **Search Logic** with a little plus sign (contained within the gray arrow) indicating that it is new and not yet delivered. Under this new component, an **Outgoing** folder contains the change set Rational Team Concert created for you with the comment **Share**. You can change this comment to something more descriptive. The comment Share was in all likelihood chosen by Rational Team Concert because the creation of the change set was a result of sharing a new Java project under a new Rational Team Concert component.

Expanding the **Share** change set, shows the individual artifacts created and a part of the new Java project.

**Associating changes to Work Items**

To associate the change set **Share** to **Work Item #4: Provide faceted search capabilities**:

1. In the **Pending Changes** view, select the **Share** change set, right-click, then select **Related Artifacts > Associate Work Item**...
2. The Select Work Items dialog opens.
3. In the Containing id or text: box, type 4, which is the Work Item that you want to add to the list.
4. The desired Work Item should show up in the list, select it, and click OK.

Figure 6 shows that the change set in the Pending Changes view now shows the id and description of Work Item #4 prefixing the comment Share. This indicates that the desired association between the change set and the Work Item has taken effect.

**Figure 6. Association of a change set to a Work Item**

More work, more changes!

The work required for Work Item #4 is not complete. You still need to create a new Java package and a class before completing the development effort on this Work Item.

1. In the Packages Explorer view, select the JKESearchLogic project, right click and select New > Class.
2. In the New Java Class dialog, enter org.foo for Package field and SearchTask for the Name field.
3. Click Finish.

Figure 7 shows the Pending Changes view which now lists three separate changes to reflect the creation of the new Java class, SearchTask which was created under the new package org.foo. These changes show up under a folder titled Unresolved.

You have the choice of:

- Checking in these changes to your existing change set as part the work for Work Item #4.
- Checking in these changes under an entirely new change set (associated to a different Work Item).

Now select and check in each of these three changes individually or altogether.
4. To check-in all of the changes together under the change set for Work Item #4, in the Pending Changes view, select and right-click the Unresolved folder.
5. Select Check-in > 4: Provide Faceted search capabilities – Share, as shown in Figure 8.

The work required for Work Item #4 is now complete. After you deliver the change set for Work Item #4, other developers on your team can use the new Rational Team Concert component created to create more Java artifacts and perhaps elaborate on the ones you created for Work Item #4.

Incorporating changes from team members

As you carry out your development through assigned Work Items, the Pending Changes view notifies you when the change sets delivered by your peers are available. It is a good practice to accept these change sets to incorporate them into your Eclipse development projects. Doing this ensures your work remains compatible with and leverages the latest changes from the work of your peers. The Incoming Change Sets topic in the Rational Team Concert Knowledge Center provides more background on accepting incoming change sets. Request a Personal Build before you deliver your work and make it available to your team, you'll want to thoroughly unit test your changes, perform static code analysis, and perhaps carry out performance testing to ensure the quality of your changes and the code base remains high.
Another best practice in a continuous integration DevOps environment is to request a personal build that includes your undelivered change sets to ensure that once you do deliver the team's integration builds do not break. Such breaks impede the overall team's development and sprint deadlines. A personal build runs in isolation from the team's integration build, but uses the team's most current code base along with all of your checked-in, but undelivered change sets. A successful personal build is the best way to know that your changes will not adversely affect the team's integration build once you deliver your change sets. You'll find the steps to request a build in the Knowledge Center.

**Deliver work!**

After arduous coding, careful debugging, and extensive testing, the change set for Work Item #4 is ready for delivery!

1. From the Pending Changes view, select the change set 4: Provide faceted search capabilities under Search Logic > Outgoing folder. Right-click and select Deliver.
2. In the Component Additions dialog box, select the Deliver component additions/removals as well as outgoing change sets and baselines radio button.
3. Click OK.

The change set delivery is complete; you can mark the Work Item as resolved and move on to your next Work Item assignment!

**Conclusion**

In this article, you used Rational Team Concert, to share a new Eclipse Java project, package, and class into a newly created Team component, checked in changes under a change set, associated a change set to a Work Item, and finally delivered the completed work to the team repository.

**Acknowledgements**

The author thanks Cheng-Yee Lin and Guillermo A. Hurtado for their thoughtful review of this article.
Related topics

- More about Rational Team Concert:
  - Learn tips and tricks in the developerWorks published Rational Team Concert articles.
  - Check out the product overview page where you'll find the features, system requirements, and more.
  - Check the Rational Team Concert page on Jazz.net.
- Improve your skills. Check the Rational training and certification catalog, which includes many types of courses on a wide range of topics.
- Download Rational Team Concert from Jazz.net and try it free on up to 10 developers for as long as you want (requires registration).

© Copyright IBM Corporation 2015
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)