

Hands-on Lab 5:

Adding New Stages to the Pipeline



Table of Contents

Objectives	1
Prerequisites	1
Time	1
Exercise 1: Adding New Stages to the Pipeline	2
Task 1: Create the Build Definition for the Exploratory Testing Stage	3
Task 2: Configure the Pipeline to Automatically Trigger the Exploratory Testing Stage	7
Task 3: Run and Monitor the Exploratory Test Stage	8
Using the Pipeline Test Stages	9
Summary	9
Copyright.....	9

Objectives

This HOL demonstrates how to add new stages to the continuous delivery pipeline that you constructed in the previous labs.

Prerequisites

To complete this lab, you must have completed the implementation of the continuous delivery pipeline.

Time

You should be able to complete this lab in 30 minutes.

Exercise 1: Adding New Stages to the Pipeline

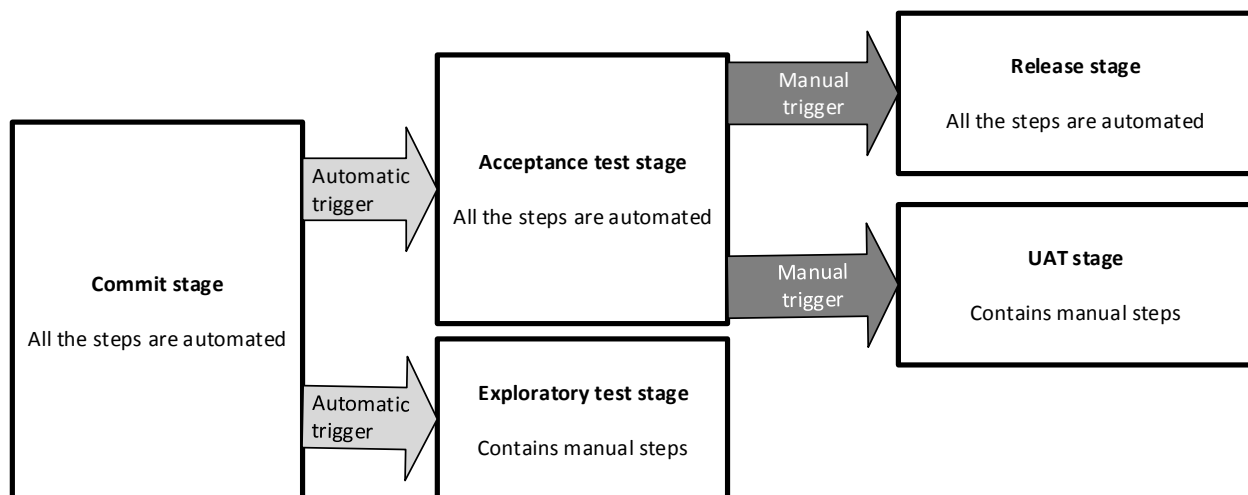
In this exercise , you learn how to add a new stage to the continuous delivery pipeline that you constructed in the previous labs. By adding new stages, you can improve the quality of your software. New stages allow you to add new validations to the release process.

Stages can be classified by their trigger and by whether the steps they contain are automated and/or manual. The four possibilities are shown in the following table.

Trigger	Steps	Examples
Automatic	Automated	The commit stage and the acceptance test stage in the continuous delivery pipeline you implemented.
Automatic	Automated and/or manual steps	An exploratory testing stage that is triggered automatically after the commit stage stage succeeds. The deployment step is done automatically but the exploratory testing is done manually.
Manual	Automated	A release stage that can be manually triggered when necessary. The deployment step is done automatically. This is the only step the stage contains.
Manual	Automated and/or manual steps	A UAT stage that can be manually triggered when necessary. The deployment is done automatically, after the stage is triggered. The UATs are manual.

In this exercise, you add an exploratory test stage, which is an example of an automatically triggered stage that contains both automated and manual steps. The stage automatically deploys the application to the target environment if the commit stage succeeds. After that, the testers complete the stage by manually performing exploratory tests.

This is how the pipeline will look after adding the Exploratory Test Stage:

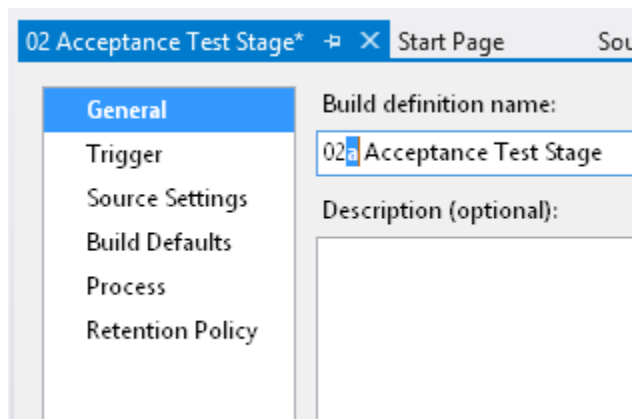
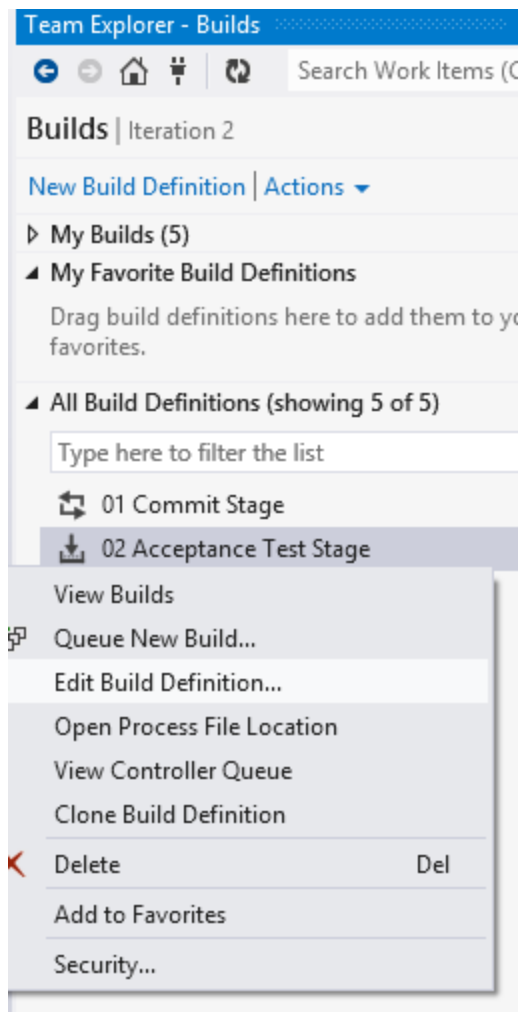


NOTE: For the sake of simplicity, this lab uses the same environments that were used in the previous labs. An improvement would be to add a new environment that is dedicated to exploratory testing. This would allow the pipeline to run multiple parallel stages until it reaches the release stage. Also, there would be fewer problems with timeouts that can occur when one stage is blocked because another is already using the environment.

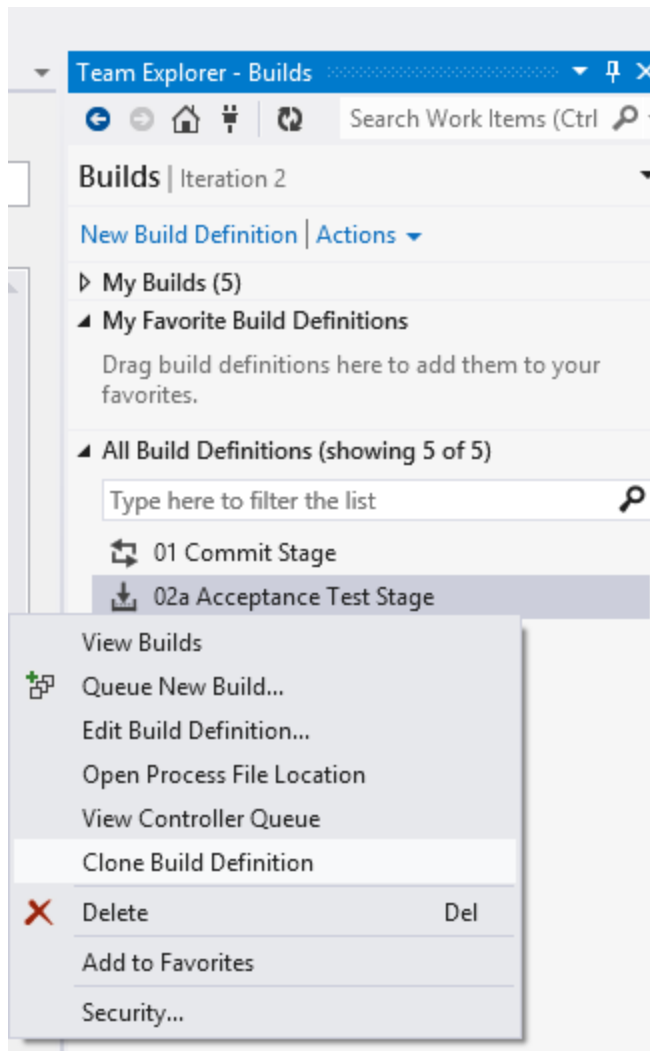
Task 1: Create the Build Definition for the Exploratory Testing Stage

The exploratory testing stage is very similar to the acceptance test stage. It is automatically triggered and it performs an automated deployment using the same scripts and the same environment. The difference is that it has no automated tests, apart from BVTs that validate the deployment. Tests that validate the application are performed manually. Because of its similarity to the acceptance test stage that you've already implemented, you can base the exploratory testing stage on the **CDPipelineGenericStageProcessTemplate** that you customized in the orchestration HOL. In fact, you can use the acceptance test stage build definition as the basis of the new stage.

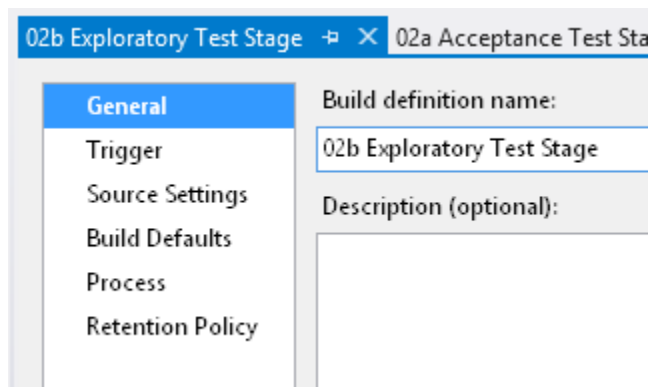
1. From **Team Explorer – Builds**, open the **02 Acceptance Test Stage** build definition. Right-click on it and select **Edit Build Definition**. Rename the build definition **02a Acceptance Test Stage**. Save the change.



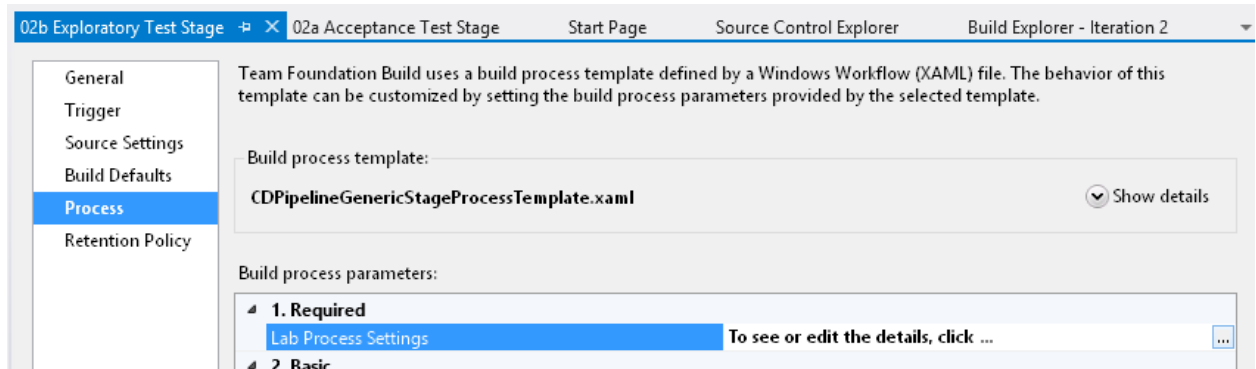
2. Clone the **02a Acceptance Test Stage** build definition.



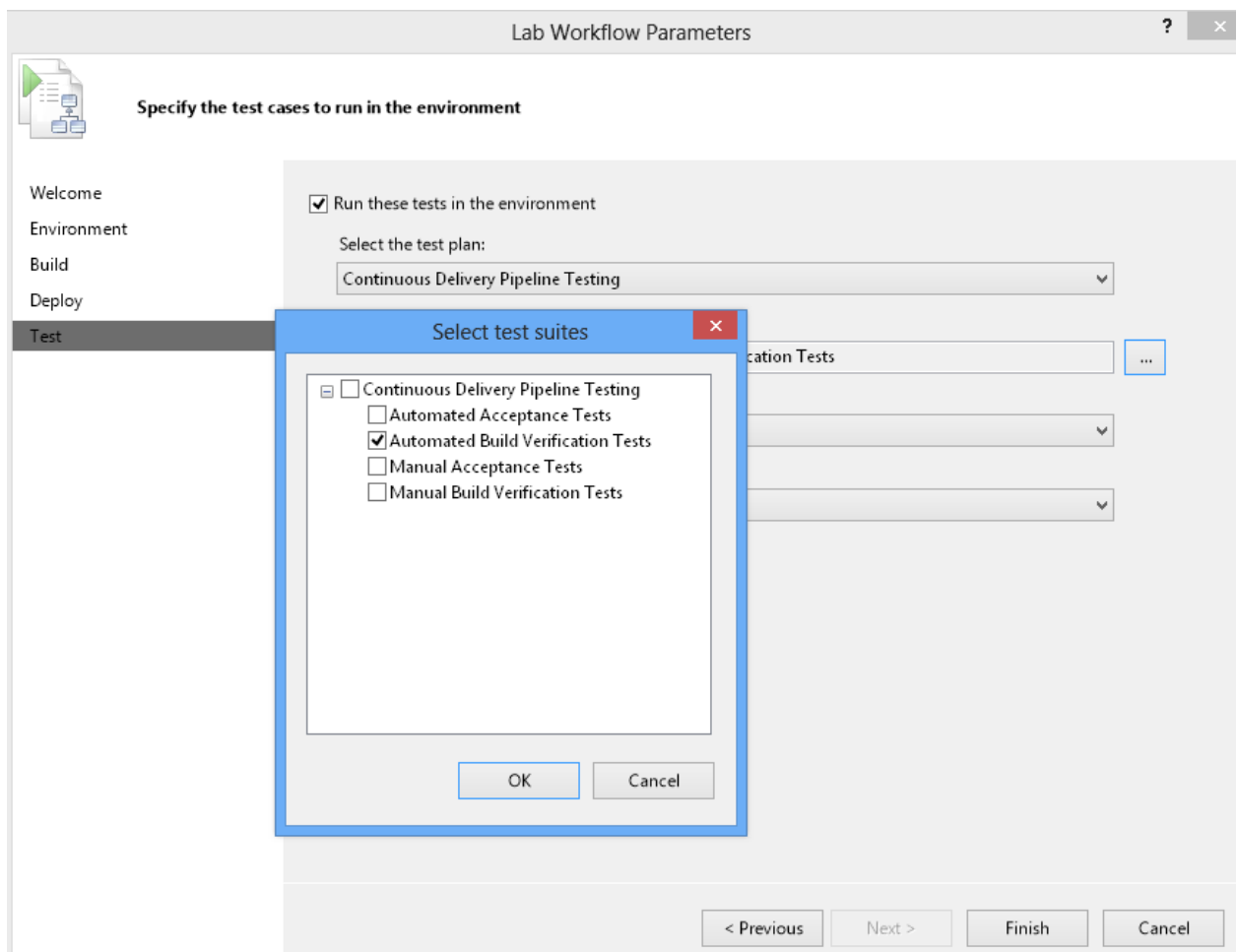
3. Change the name of the cloned build definition to **02b Exploratory Test Stage**.



4. Select the **Process** tab. Open the **Lab Process Settings** by clicking on the ellipsis (...).



5. Select the **Test** tab. Leave **Automated Build Verification Tests** selected. Clear any other selections.

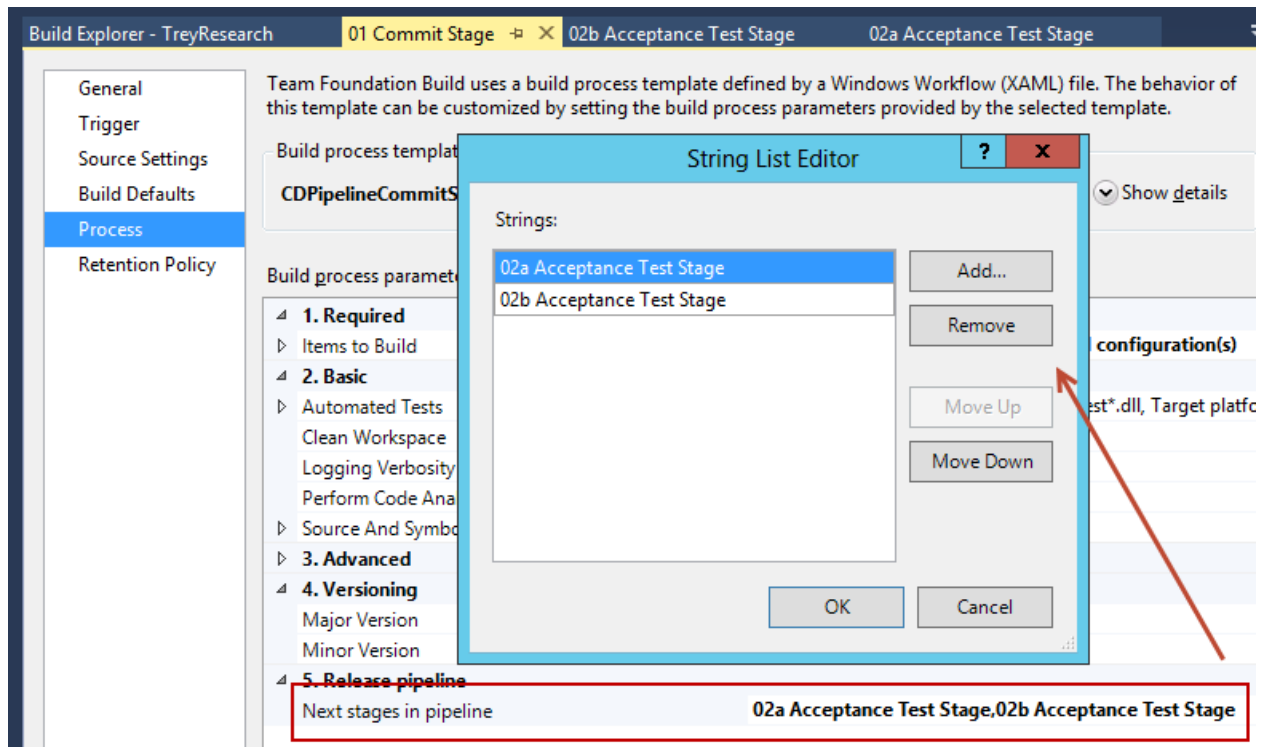


6. Click **Finish**. Save the build definition.

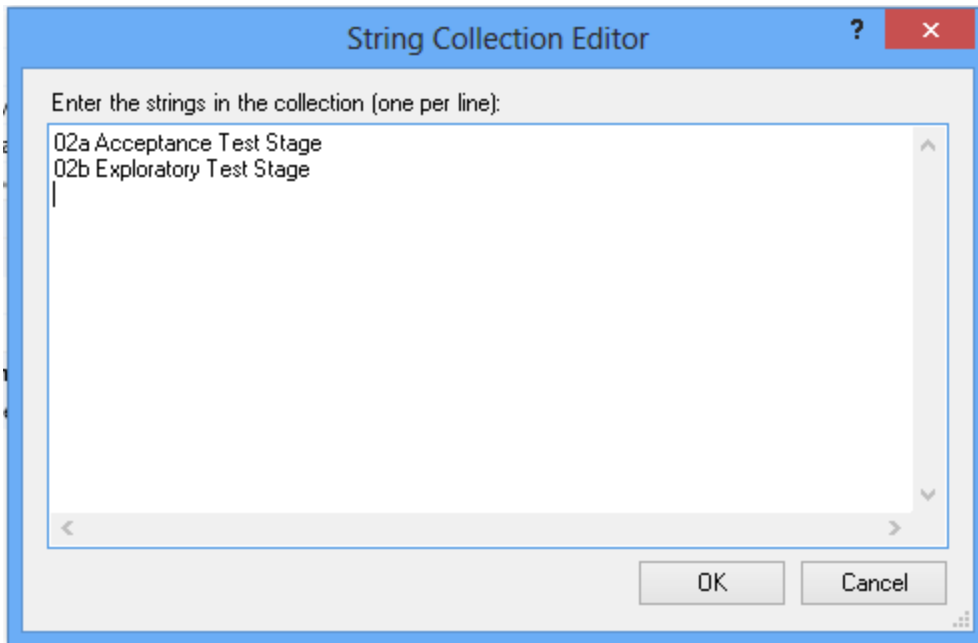
Task 2: Configure the Pipeline to Automatically Trigger the Exploratory Testing Stage

The exploratory testing stage is automatically triggered when the commit stage finishes successfully. This means you must modify the commit stage build definition so that it knows to trigger the new stage.

1. From **Team Explorer – Builds**, open the **01 Commit Stage** build definition. Right-click on it and select **Edit Build Definition**.
2. Select the **Process** tab. Go to **section 5. Release pipeline** of the Build process parameters. Select **Next stages in pipeline**. Click the ellipsis (...).



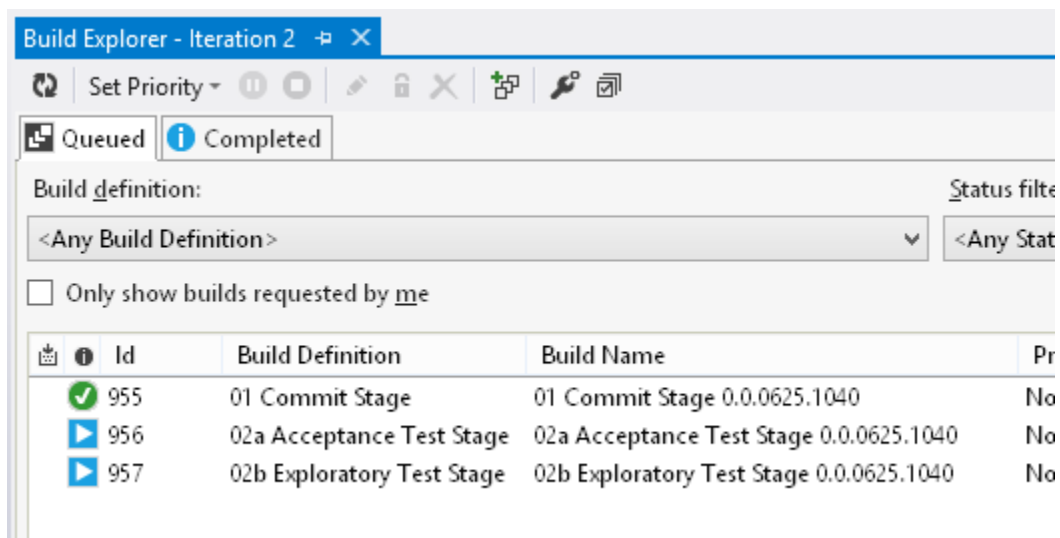
3. The **String Collection Editor** opens. Change the name of the acceptance test stage to **02a Acceptance Test Stage**. Add a new line with the name **02b Exploratory Test Stage**. Click **OK**. Save the build definition.



Task 3: Run and Monitor the Exploratory Test Stage

In this task you trigger a new pipeline instance and monitor it to see how it works with the new stage.

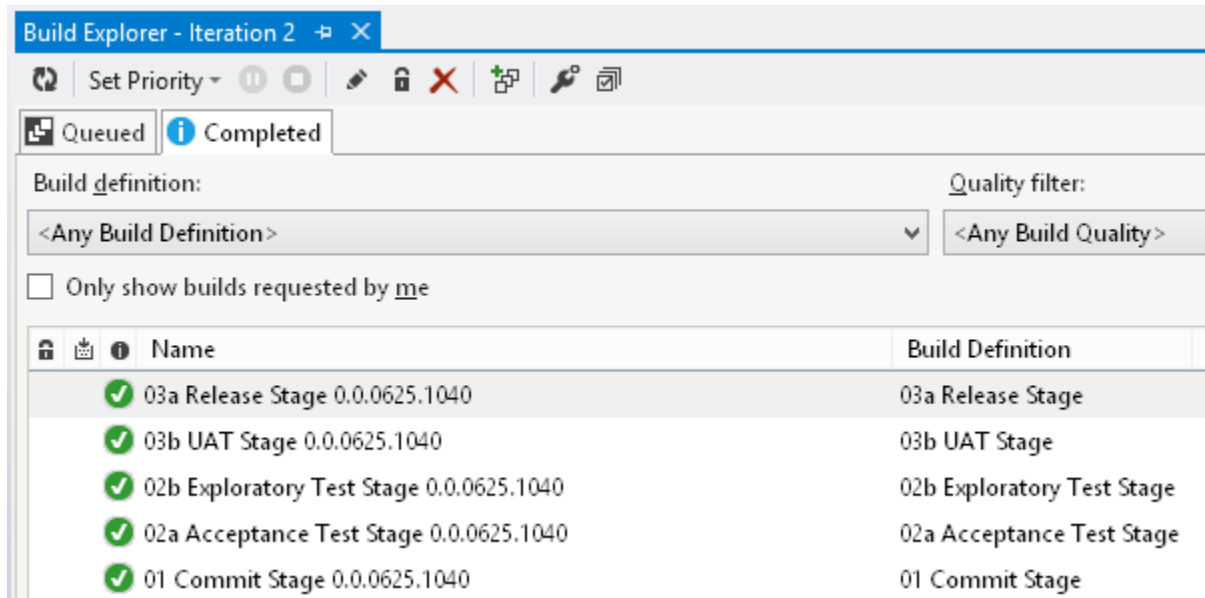
1. Trigger an instance of the pipeline by queuing the **01 Commit Stage** build definition. Make sure that Build Explorer is open so that you can monitor the progress of the pipeline instance.
2. After the commit stage finishes successfully, both the **02a Acceptance Test Stage** and the **02b Exploratory Test Stage** should be triggered.



Using the Pipeline Test Stages

Once the exploratory test stage is triggered, you can immediately perform tests on any changes to the application because it is already deployed to the target environment and is ready to be used. You can also run the UAT stage so that users can begin to perform user acceptance tests. When all the tests are done, you can, if you want, manually trigger the release stage so that the fully tested application is available to end users.

The following screenshot shows a pipeline instance that has successfully completed all its stages.



Summary

In this HOL you added an exploratory test stage to the continuous delivery pipeline that you implemented in the previous labs. This stage is automatically triggered by the commit stage. The new stage has one automated step, which is to deploy the application to the target environment. Tests that validate the application are performed manually.

To create the new stage, you adapted the existing acceptance test stage by modifying its build definition. You also modified the commit stage build definition to include the exploratory test stage in the list of automatically triggered stages.

Finally, you triggered an instance of the new pipeline and used Build Explorer to monitor its progress.

Copyright

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet website references, may change without notice. You bear the risk of using it. Some

examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

© 2014 Microsoft. All rights reserved.

Microsoft, Windows, Windows Server, Windows Vista, Windows PowerShell, Silverlight, Expression, Expression Blend, MSDN, IntelliSense, IntelliTrace, Internet Explorer, SQL Azure, SQL Server, Visual C#, Visual C++, Visual Basic, and Visual Studio are trademarks of the Microsoft group of companies.

All other trademarks are the property of their respective owners.