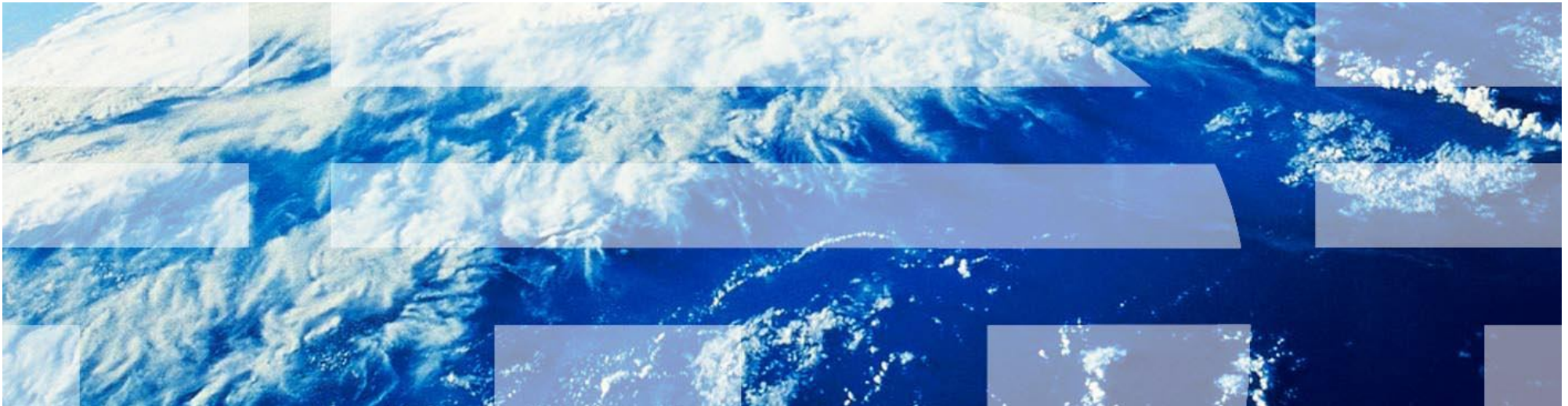


Module 4 – Completing a story



Agenda

Introduction to Collaborative Lifecycle Management

Lab Overview

Module 1: Aligning plans

Module 2: Responding to a change request

Module 3: Planning the sprint

Module 4: Completing a story

~~Module 5: Testing and fixing a story~~

Module 6: Stabilizing the sprint

Session Summary

Objectives

Explore how IBM Rational Quality Manager can be used to create test implementations (scripts) and associate them with test cases.

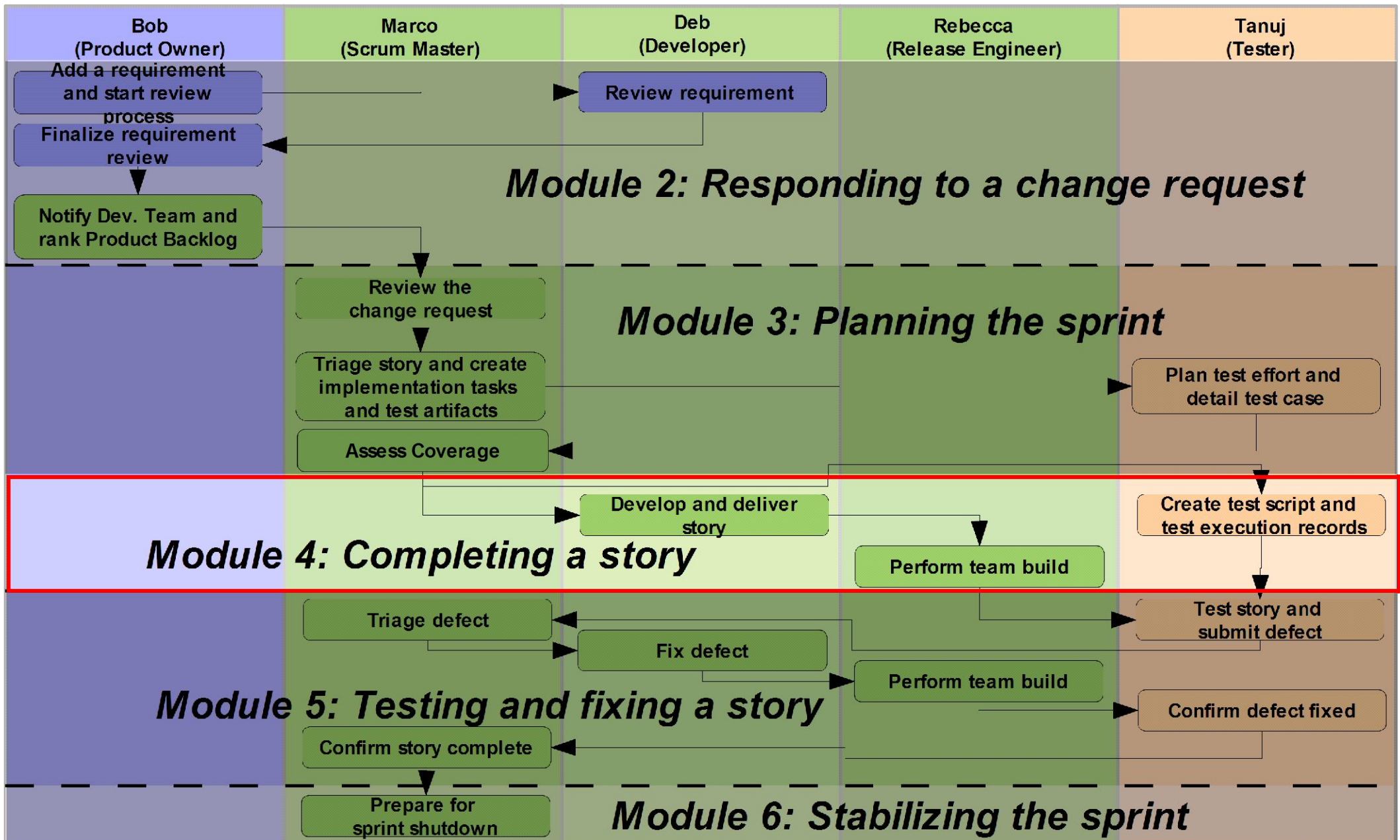
Explore the IBM Rational Team Concert Eclipse client from a developer's perspective.

Understand how to use IBM Rational Team Concert to accept work, complete development tasks and deliver updated work to the team.

Explore the team build features of IBM Rational Team Concert.

Explore how all team members can monitor the status of team builds.

Lab Sequence of Events



Lab 4 Scenario

We will play three different roles in this lab!

As Tanuj, a tester, we will use IBM Rational Quality Manager to create and edit new test scripts and test execution records for the test case you added in the previous lab.

As Deb, a developer, we will use the IBM Rational Team Concert Eclipse client to plan your work, complete your development tasks, execute an personal build, and deliver your updates to the team.

As Rebecca, the release engineer, we will use the IBM Rational Team Concert Eclipse client to execute an integration build on the team's build server.

Lab 4 Concepts Learned

IBM Rational Quality Manager provides a highly descriptive manual testing facility that provides the tester with the right level of detail required to execute the test correctly.

IBM Rational Quality Manager test scripts implement test cases that can be linked to the drivers for those test cases (in this case, stories).

Jazz Source Control provides private repository workspaces to track and back up your changes before you share them with the team using a stream for integration.

A change set is the fundamental unit of change and collaboration in your team environment. Change sets are migrated between streams via two operations: accepting and delivering.

Developers can run "personal builds" on the team's build server to ensure that the code they see in their workspace successfully builds using the team's build process. Ensuring compilation in the integrated development environment (IDE) isn't always enough.

Team members can request a "team build" that will grab the latest code on the team's integration stream and build it on the team build server.

Every team member has access to build data from team builds. This promotes communication and collaboration among the contributors – on local or remote sites.

Questions

