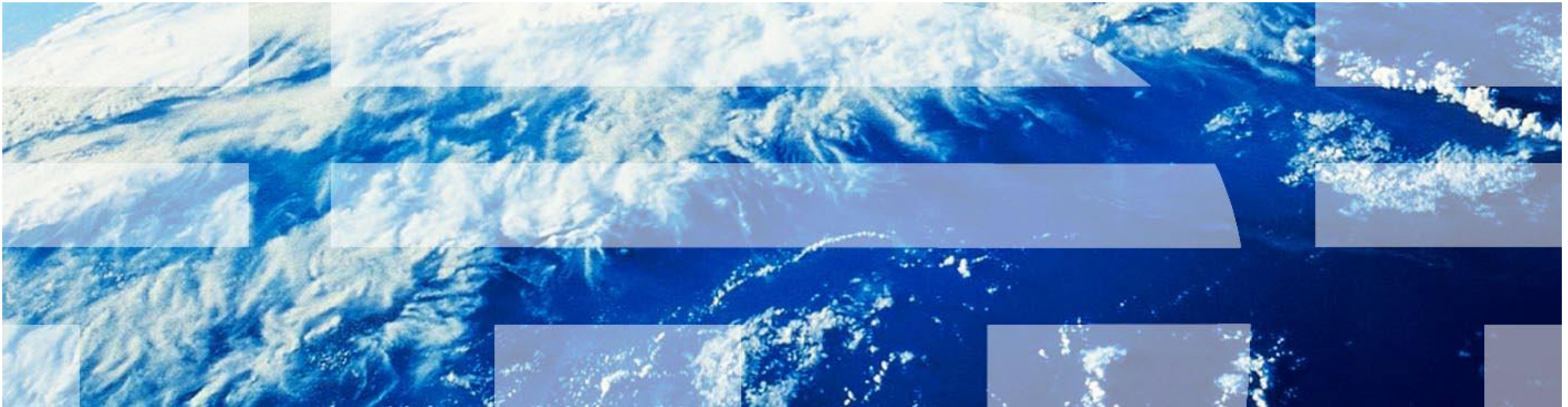


# Module 3 – Planning the sprint



# 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

Further explore the IBM Rational Team Concert (RTC) Web interface as well as dashboards and viewlets.

Explore how IBM Rational Team Concert enables agile teams to manage sprint/iteration plans and manage tasks.

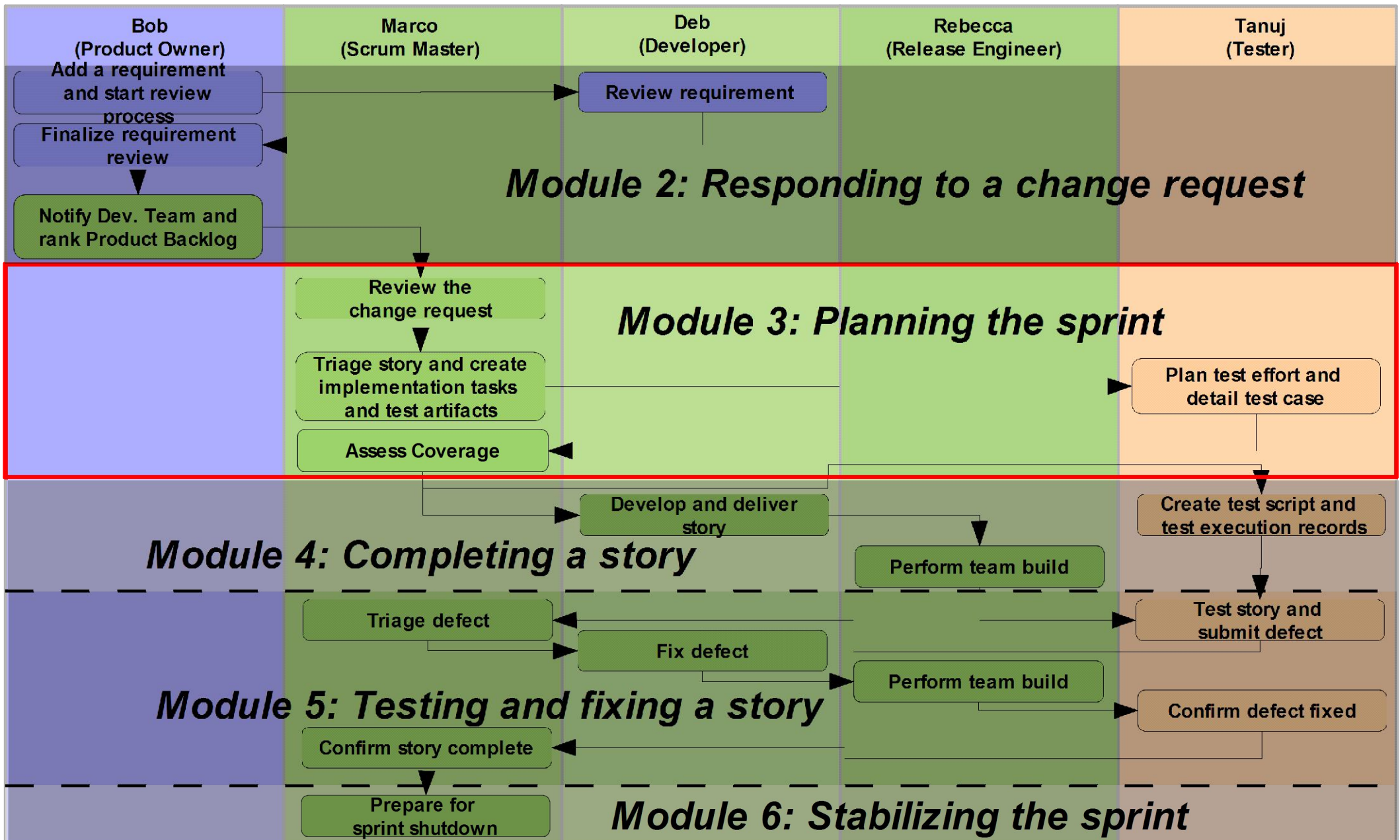
Explore how IBM Rational Team Concert can be used to facilitate a sprint planning meeting.

Create a new test case directly from IBM Rational Team Concert associated with a story.

Explore how IBM Rational Quality Manager can be used to detail and organize test cases and plan test effort.

Explore how to assess plan coverage from with IBM Rational Team Concert.

# Lab Sequence of Events



## Lab 3 Scenario

We will play two different roles in this lab!

As Marco, the Scrum Master and development lead, we will use the IBM Rational Team Concert Web interface to review the Product Backlog, triage the new story to the current sprint/iteration, and create the implementation tasks and associated test case.

As Tanuj, a tester, we will use IBM Rational Quality Manager to review and detail the newly create test case and plan the test effort.

As Marco again, we will assess plan coverage using IBM Rational Team Concert.

## Lab 3 Concepts Learned

IBM Rational Team Concert makes it easy to manage a project's development plan.

Stories and related implementation tasks defined in IBM Rational Team Concert (RTC) can be used to create and link to new or existing test case in IBM Rational Quality Manager (RQM).

IBM Rational Quality Manager helps create and manage test plan and test cases linked with development stories.

IBM Rational Requirements Composer, IBM Rational Team Concert and IBM Rational Quality Manager provides full visibility of CLM relationships across requirements, development and test artifacts.

# Questions

