Test Automation & Advanced Selenium

Lesson 4: Selenium 2.0 – Web Driver

Lesson Objectives

- Introduction To Web Driver
- Web Driver Vs Selenium RC Vs Selenium IDE
- Benefits of Web Driver over Selenium IDE and RC
- Limitations of Web Driver

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 2

4.1: Selenium 2.0 – Web Driver

Introduction To Web Driver

- Web automation framework that allows you to execute your tests against different browsers, not just Firefox (unlike Selenium IDE).
- Provides a simpler, more concise programming interface
- Selenium-WebDriver was developed to better support dynamic web pages where elements of a page may change without the page itself being reloaded
- Supply a well-designed object-oriented API that provides improved support for modern advanced web-app testing problems.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 3

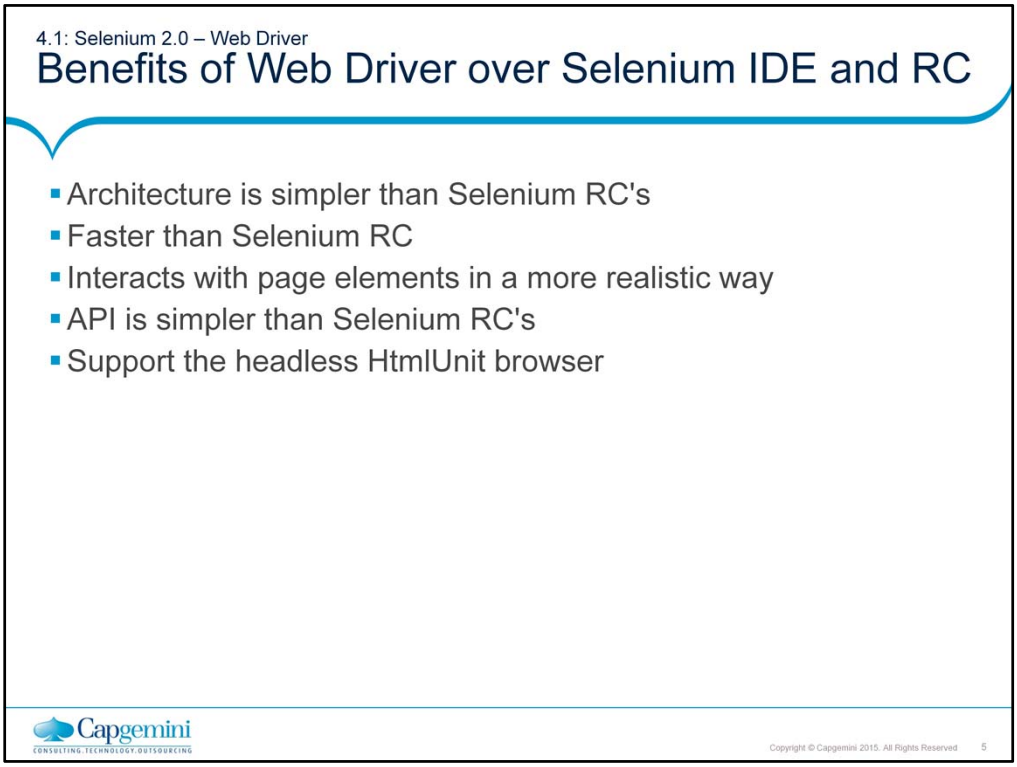
4.1: Selenium 2.0 – Web Driver

Web Driver Vs Selenium RC Vs Selenium IDE

WebDriver	Selenium RC	Selenium IDE
▪ Supports all browsers like Firefox, IE, Chrome, Safari, Opera etc. ▪ Supports Record and playback ▪ Doesn't required to start server before executing the test script ▪ Interacts natively with browser application ▪ As compared to RC, it is bit complex and large API.	▪ Supports all browsers like Firefox, IE, Chrome, Safari, Opera etc. ▪ Doesn't supports Record and playback ▪ Required to start server before executing the test script. ▪ Core engine is JavaScript based ▪ It is easy and small API	▪ Only works in Mozilla browser ▪ Doesn't supports Record and playback ▪ Doesn't required to start server before executing the test script. ▪ Core engine is JavaScript based ▪ Very simple to use

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 4



1. Architecture
WebDriver's architecture is simpler than Selenium RC's.

It controls the browser from the OS level
All you need are your programming language's IDE (which contains your Selenium commands) and a browser.
You first need to launch a separate application called Selenium Remote Control (RC) Server before you can start testing
The Selenium RC Server acts as a "middleman" between your Selenium commands and your browser
When you begin testing, Selenium RC Server "injects" a Javascript program called Selenium Core into the browser.
Once injected, Selenium Core will start receiving instructions relayed by the RC Server from your test program.
When the instructions are received, Selenium Core will execute them as Javascript commands.
The browser will obey the instructions of Selenium Core, and will relay its response to the RC Server.
The RC Server will receive the response of the browser and then display the results to you.
RC Server will fetch the next instruction from your test script to repeat the whole cycle.

2. Speed
WebDriver is faster than Selenium RC since it speaks directly to the browser uses the browser's own engine to control it.

- Selenium RC is slower since it uses a Javascript program called Selenium Core. This Selenium Core is the one that directly controls the browser, not you.

3. Real-life Interaction
WebDriver interacts with page elements in a more realistic way. For example, if you have a disabled text box on a page you were testing, WebDriver really cannot enter any value in it just as how a real person cannot.

- Selenium Core, just like other Javascript codes, can access disabled elements. In the past, Selenium testers complain that Selenium Core was able to enter values to a disabled text box in their tests. Differences in API

4. API
Selenium RC's API is more matured but contains redundancies and often confusing commands.

- For example, most of the time, testers are confused whether to use type or typeKeys; or whether to use click, mouseDown, or mouseDownAt. Worse, different browsers interpret each of these commands in different ways tool
- WebDriver's API is simpler than Selenium RC's. It does not contain redundant and confusing commands.

5. Browser Support
WebDriver can support the headless HtmlUnit browser

- HtmlUnit is termed as "headless" because it is an invisible browser - it is GUI-less.
- It is a very fast browser because no time is spent in waiting for page elements to load. This accelerates your test execution cycles.
- Since it is invisible to the user, it can only be controlled through automated means.
- Selenium RC cannot support the headless HtmlUnit browser. It needs a real, visible browser to operate on.

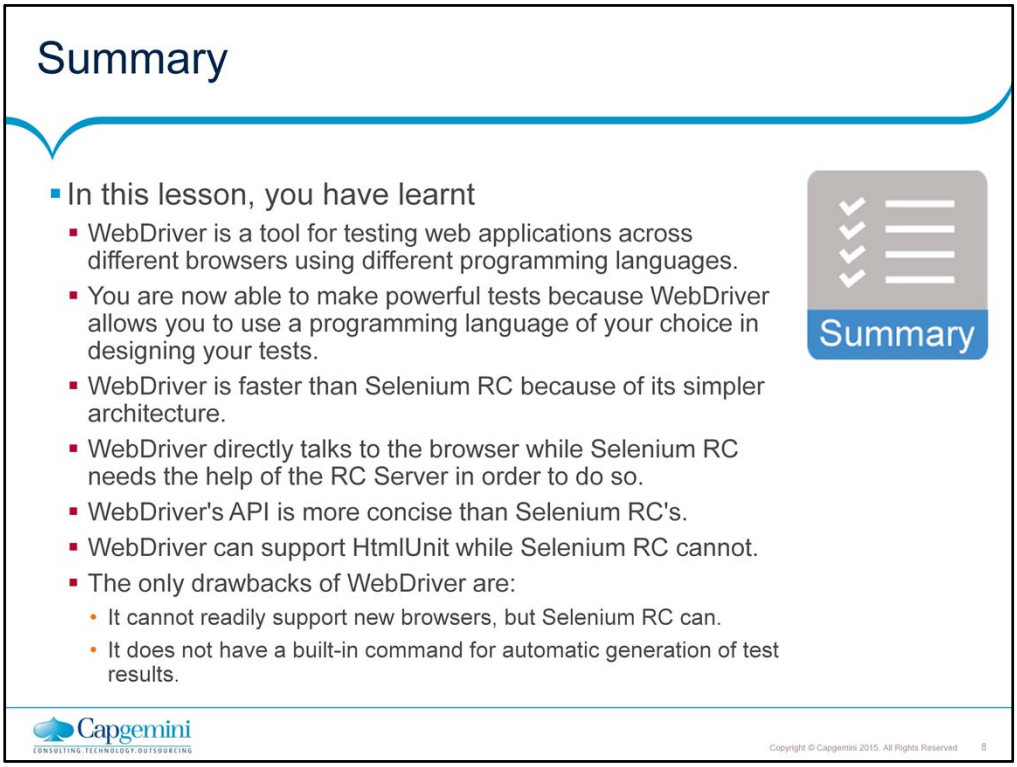
4.1: Selenium 2.0 – Web Driver

Limitations of Web Driver

- Cannot Readily Support New Browsers
  - Since, WebDriver operates on the OS level
  - Different browsers communicate with the OS in different ways.
  - If a new browser comes out, it may have a different process of communicating with the OS as compared to other browsers
- Has no built-in command that automatically generates a Test Results File
  - Have to rely on your IDE's output window, or design the report yourself using the capabilities of your programming language and store it as text, html, etc.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 7



Add the notes here.

Review Question

- Question 1
  - Select which is NOT a feature of Web Driver
  - Architecture is simpler
  - Faster
  - API is complex
  - Supports all the browsers

- Question 2: True/False
  - WebDriver can Readily Support New Browsers.
- Question 3: Fill in the Blanks
  - WebDriver can support _____while Selenium RC cannot.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 9