Selenium WebDriver Step by Step

Build Enterprise-Level Test Automation Framework with Real Project

Ken Ho

About me

- 10+ years experience in software QA and Development area
- Worked in large companies like Microsoft and IBM, as well as start-up companies and medium-sized companies
- First hand experiences as Automation Framework designer and developer

In this course

- A complete enterprise-level test automation framework, including automated test execution, prioritization, report, integration with continuous integration
- Step by step guide to build this cross-browser automation framework with real-life examples
- Share the experience on how to make the framework easy-to-maintain and expansible

The Project Requirement - Test Gmail(1)

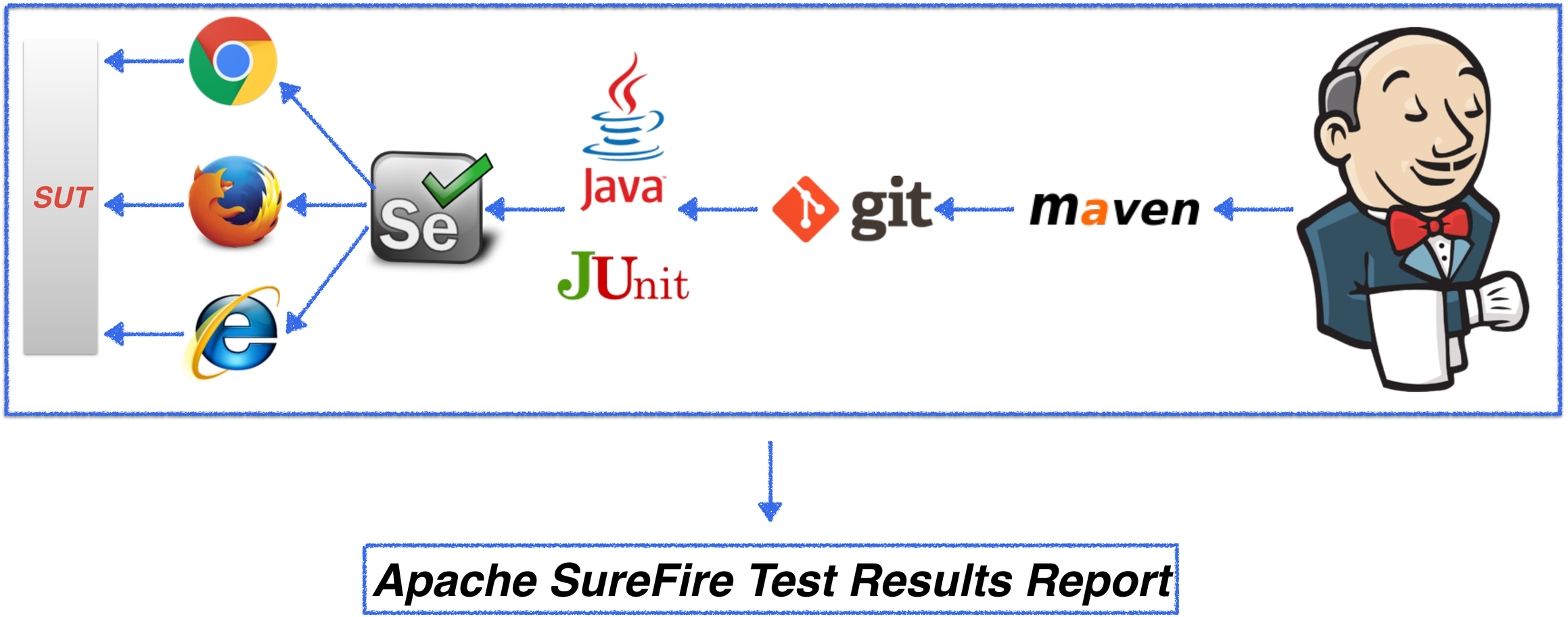
- Ensure below functionality of AUT is working well, in Firefox, Chrome, IE, Safari under Windows, MacOS, Linux
- Positive Test Scenarios:
    - Sign in + Sign out(P1)
    - Send email + Receive email(P1)
    - Send email + Receive email with attachment(P1)
    - Automatically save draft email(P1)
    - Search email(P2)
    - Send email + Delete email(P2)
- Negative Test Scenarios:
    - Send email without recipient(P2) - Should be able to remind user, email cannot be sent until fixed
    - Send email without subject(P2) - Should be able to automatically add a default subject (no subject)

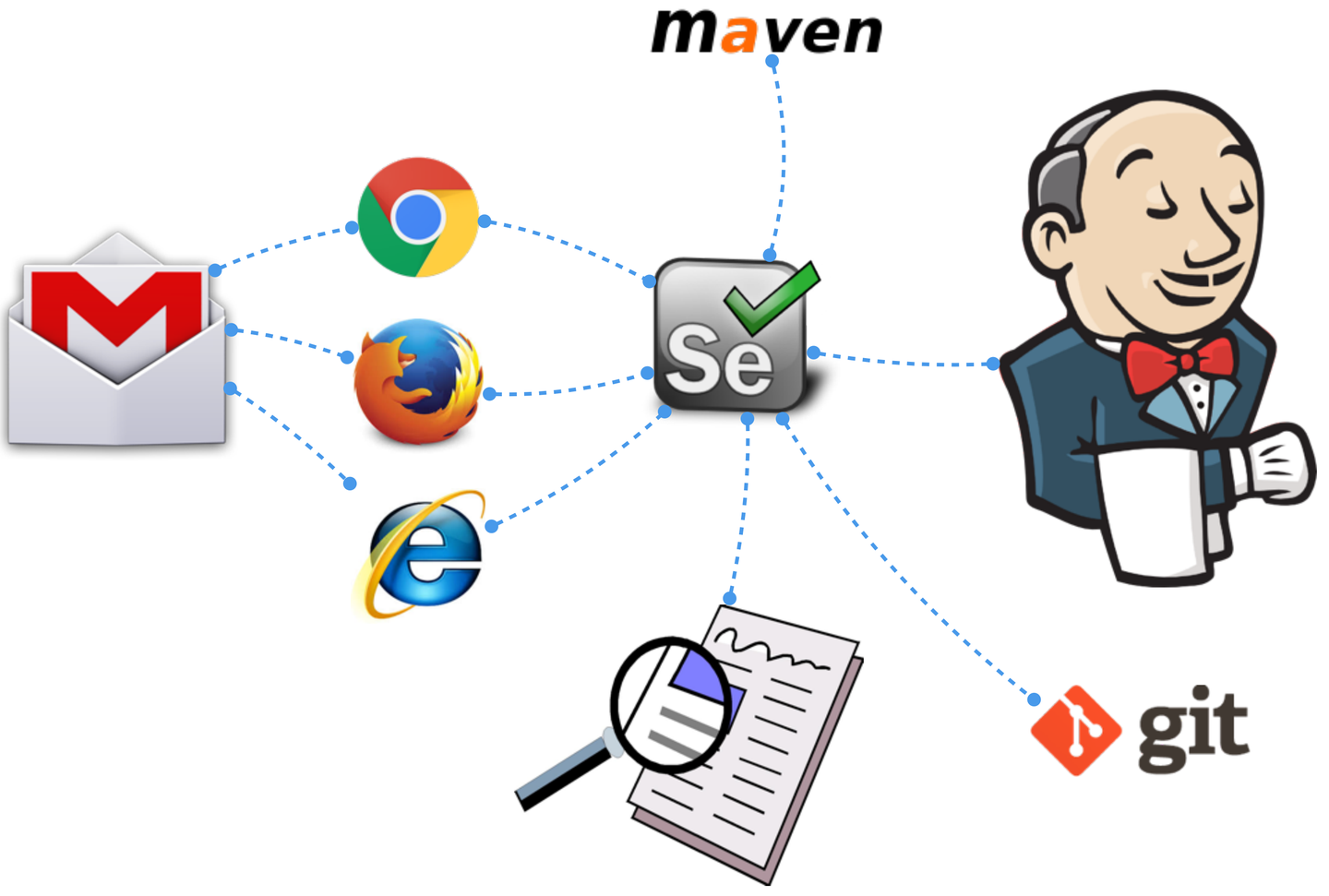
The Project Requirement - Test Gmail(2)

- Framework Business Requirement
  - Can run tests according to pre-defined priority
  - Support cross-browser testing based by re-using same test scripts
  - Trigger test automatically when there is SUT changed to support CI
  - Generate accurate test result(Pass/Fail) report
  - Check in test framework codes into VCS
  - Easy to maintain and expansible

How will you design the test framework?

Test Framework Design




maven
Se
git

Before Implementing Automation Scripts…

- Walk through a Manual Test first!
- Sign in and sign out
    - Go to Gmail website
    - Fill in username
    - Fill in password
    - click sign in
    - verify user did sign in
    - sign out
    - verified user did sign out

Create your first automation test

Get tools ready

- java jdk - http://www.oracle.com/technetwork/java/javase/downloads/index.html
- IntelliJ - https://www.jetbrains.com/idea/download/
- maven - https://maven.apache.org
- Selenium - http://www.seleniumhq.org/download/maven.jsp
- Git - install Xcode from apple store

Prepare FireFox

- Please use FireFox v31, as newer versions may not work well with Selenium WebDriver 2.43 or lower

Implementing the scripts by following test steps

- Sign in and sign out
    - Go to Gmail website
    - Fill in username
    - Fill in password
    - click sign in
    - verify user did sign in
    - sign out
    - verified user did sign out

Quick Review

- Break down test scenarios(business requirements) to detailed test steps through manual test
- Building automation test scripts from test steps
- Each step is usually a combination of:
  - Locate WebElement by using different locators
  - Interact with the WebElement
- Solved first challenge: Synchronization by using WebDriverWait(also known as, Explicit Wait)

Managing the codes with Git

Write another test case
Send email + Receive email

Send email + Receive email Test Steps

- Send email + Receive email
    - Click sign in
    - Click Compose
    - Fill in recipient
    - Fill in subject
    - Fill in email body
    - Click Send
    - Click Inbox again
    - Click email
    - Verify the email subject and email body is correct
    - Sign out

Refactor using Page Objects Design Pattern

Categorize the test cases using JUnit

Generate test report using Apache SureFire

Integrating with Jenkins

Challenge to you:
Complete the other test cases!

- Download source code here from BitBucket. Repository information is share in the platform.
- Get very familiar with different kinds of selectors beside id and name, especially CssSelector and XPath
- And you will become a more experienced QA Engineer!

Thank you for taking this course!

- Recommended further studies:
  - Different locators: Css Selector and XPath
  - Appium for Mobile Automation
  - Utilize Selenium Grid and Sauce Labs for testing in the “cloud”
  - Software Development Life Cycle related tools, such as, git, maven, Jenkins
- Contact me through the platform or appsenseca@gmail.com.

Code Update on Nov 21, 2015

- 1. Update Selenium Version to 2.48.2, to support FireFox 42
- 2. Updated a method to smartly wait for new email with specified subject to arrive
- 3. Added a method to dismiss with Alert popup
- 4. Created a new branch: mac_os_chrome_driver, to demonstrate how to include chromedriver bin in your repo, so that other users can use Chrome browser without setup the driver themselves
- 5. use the new branch by:
  - git pull
  - git checkout mac_os_chrome_driver
- 6. Hope you like it!

Start Selenium Grid

- 1. Download selenium server standalone jar file from http://www.seleniumhq.org/download/
- 2. Start Selenium Hub and Node
- 3. Update the Test Script
- 4. Run it!