Survey of Comparative Study for Automated Tools such as Selenium RC, Selenium Web Driver

Mahan Sunhare[1] Abhishek Tiwari[2]
[1,2] M.I.T., Ujjain

Abstract— In SDLC, Testing is the technique to evaluate a software to detect differences between given input and expected result. Testing evaluates the standard of the software deliverable. In other words, Testing is a process of verification and validation. Manual testing is performed by a tester carefully executing the test cases. In Automation Testing, an automation tool to execute your test case suite. The automation tool can also feed the test data into the System under Test, compare actual and expected behaviors and produce detailed test reports. Test Automation demands resources and considerable amount of investments of money. Using a test automation software, it is possible to record the test suite and play it again if it is required. No human effort is required, once the test suite is automated. Automation increases Test Coverage and speed of test. The goal of the thesis is to present a comparative analysis of automation tools such as Selenium RC, Selenium Web Driver. The aim of the thesis is to analyze two automation software testing tools and evaluate to determine their productiveness and usefulness. Selenium is a most likely automated testing suite. Selenium is developed in a way to support and promote the automation testing of behavioral outlook of web based applications and a broader range of browsers and platforms. Due to its presence in the large open source community, it has become one of the most desired tools among the testing professionals. Selenium is not just a single automated tool, rather a package of several automated testing tools.

Key words: Selenium RC, Selenium WebDriver, Test Automation

I. INTRODUCTION

Test automation used in:

- Recurrent regression testing
- Quick feedback to developers
- Support for Agile based methodologies.
- Customized defect reporting.
- Finding defects skipped by manual testing.

Selenium is a robust set of tools which supports rapid development of test automation for web-based applications. It is licensed under Apache License 2.0.

The selenium suite package consists of the Selenium Remote Control (RC), Selenium Web Driver and Selenium Grid.

II. SELENIUM REMOTE CONTROL (RC)

Selenium RC is an automation test tool that allows us for writing automated web application UI tests in any programming language along with HTTP website using any mainstream for JavaScript-enabled browser. Selenium RC has two parts.

A server which acts as a HTTP proxy for web requests automatically launches and kills browsers.

Client libraries for your favorite computer language.

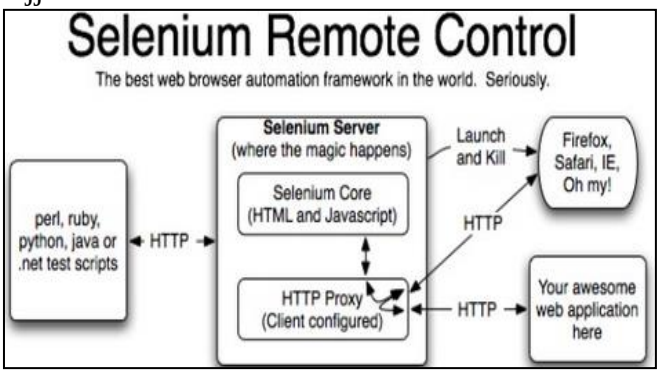

Fig. 1: Architecture of Selenium RC

Selenium RC also supports multiple languages like Java, Ruby, C#, Perl, Python etc. preferred language is Java language. Similar to language independent, Same code will work on Linux, Mac, Windows OS & Solaris so we can say Selenium RC is also a platform independent. RC is also has some limitations. Before start testing, we have to start & stop the server to execute you test.

Architecture of Selenium RC's is more complicated than selenium WebDriver.

First need to launch a separate application i.e. Selenium Remote Control (RC) Server before start testing. The Selenium RC Server plays a role as an "agent" between Selenium commands given by you and your browser. When you start testing, Selenium RC Server injects a JavaScript program i.e. called Selenium Core into the browser. After injection, Selenium Core will start getting the instructions relayed by the RC Server from the test program.

After getting the instructions, Selenium Core would execute them as JavaScript commands. The browser would follow the instructions of Selenium Core, and would relay its output (response) to the RC Server. The RC Server will get the response of the browser then display the results to your Server will get the next instruction from your test script and repeat the whole cycle.

Selenium RC is slower since it uses a JavaScript program i.e. called Selenium Core. Selenium Core directly controls the browser, not you.

Selenium RC's API is more matured but contains some redundancies and also having few confusing commands.

For example, most of the time, testers are in dilemma whether to use type or type Keys; or whether to use click, mouse Down, or mouse Down at. Worse, different browsers interpret each of these commands in different ways too!

III. MERITS AND DEMERITS OF SELENIUM REMOTE CONTROL

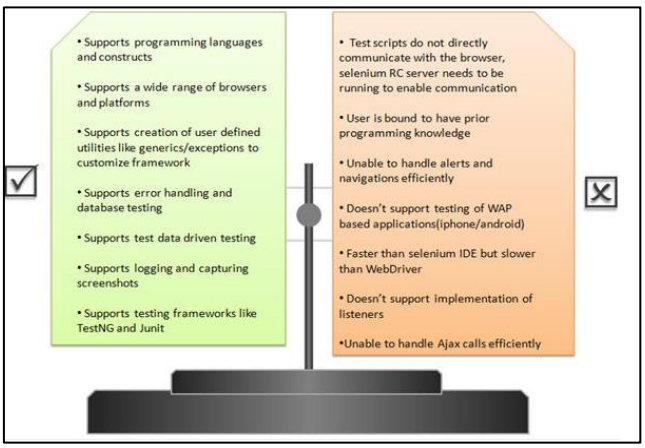

Fig. 2: Advantages & Disadvantages of Selenium Remote Control

IV. SELENIUM WEBDRIVER

WebDriver is an automation tool for web application testing, and in specific to verify that they work as expected.

Goal of WebDriver is to provide a friendly API that's easy to understand and explore, easier to use than the Selenium-RC (1.0) API, which will support to make your tests easier to maintain and read.

It's not tied to any particular web application based test framework, so it can be used similarly well in a unit testing or from a simple old "main" method.

Selenium WebDriver supports the following browsers:

- Chrome,
- Internet Explorer (IE),
- Firefox,
- Opera 11.5+,
- Html Unit.
- Android – phones and tablets (devices & emulators)
- iOS phones (devices & emulators) and for tablets (devices & emulators)

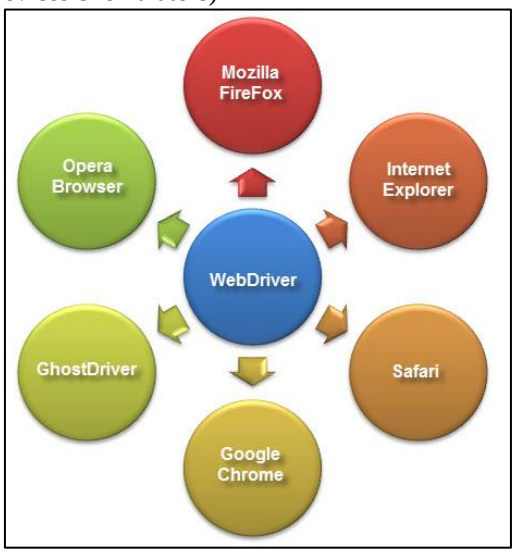

Fig. 3: Browsers

As a result, Selenium WebDriver is a most useful automated testing tool was developed called Selenium 2.0.

WebDriver is a purely object oriented testing framework.

WebDriver supports following programming languages

- Java
- .Net
- PHP
- Python
- Perl
- Ruby

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

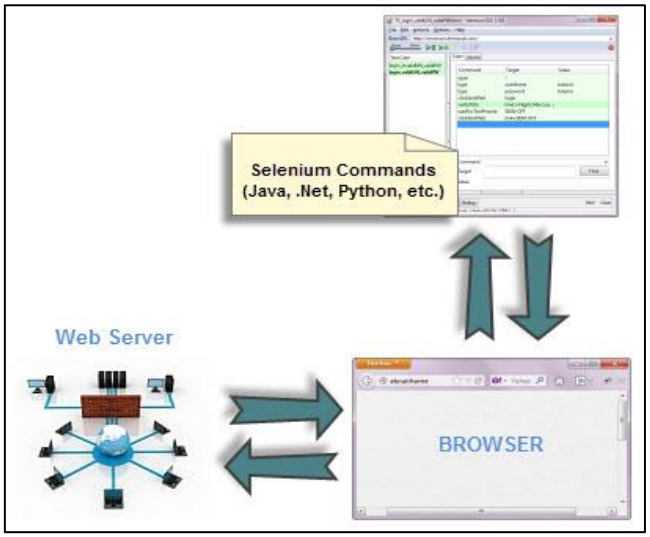

Fig. 4: Advantages & Disadvantages of Selenium Remote Control

V. MERITS AND DEMERITS OF SELENIUM REMOTE CONTROL

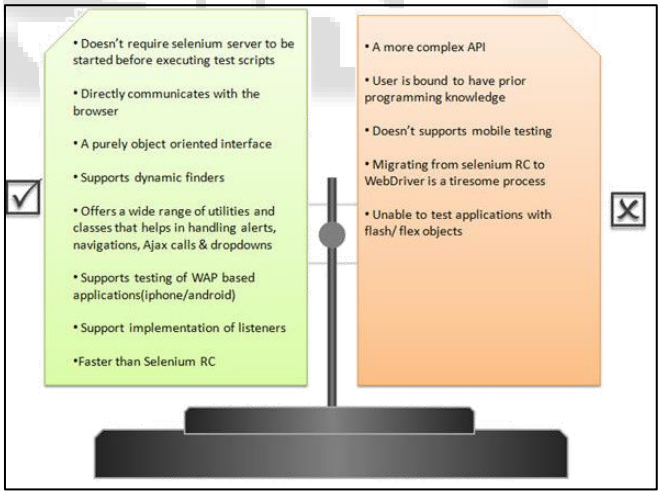

Fig. 5: Advantages & Disadvantages of Selenium Remote Control

VI. RELATED WORK

In SDLC, test automation is used to control the execution of tests (test cases) and the comparison of actual outcomes to expected outcomes. Test automation can be used to automate some repetitive but necessary tasks in a formalized testing process already in place, or add some additional testing that would be difficult to perform manually [1].

With the age of internet technology, Web application becomes more and more complex. The reliability and quality of the product is a matter of concern for the organizations. In Web application testing, especially

the regression testing is more difficult process for manual testers [2], [3].

Software developers tend to build "beautiful" pages but don't care about accessibility. Microsoft Internet Explorer is used by the majority of the company so that the pages are viewed only by this browser(IE), ignoring that in other browsers their pages look poorly, or (in some cases) some features are not visible [4].

Research is required to help focusing on Browser compatibility issue for consistent look and feel across the browsers. Therefore, it becomes vital to carry out research by providing tools and mechanisms that would help towards Browser based testing in a faster way using Selenium Web Driver as compared to Selenium RC. This paper intends to give a contribution in this direction by proposing browser based testing tool like Selenium Web Driver.

VII. CONCLUSION

In this paper, we presented a comparative study for automated tools such as available in market in Selenium RC, Selenium Web Driver. In our next paper, we will propose an approach for multiple browser support using Web Driver which is faster than selenium RC.

REFERENCES

[1] A. Kolawa, D. Huizinga. (2007). Automated Defect Prevention: Best Practice in Software Management. IEEE Computer Society Press. ISBN 0-470-04212-5.
[2] F. Coda, C. Ghezzi, G. Vigna et al. "A Software Engineering concept towards Web Based Development". 'The Ninth International workshop' on 'Software Design and Specification', Ise-Shirna Japan: IEEE 'Computer Society Press', April 16-18, 1998, pp. 8-17.
[3] J. T. Yang, J. L. Huang, F. J. Wang, and W. C. Chu, 'An Object-Oriented Framework Supporting Web Based Testing'. IEEE twenty third Annual International Computer S/W and Application Conference, USA: IEEE Computer Society Press, October 2000, pp. 122-127.
[4] M Kirchner, A standard approach for Web Based Testing for the Pages Accessibility. The 5th IEEE International Workshop on Web Site Evolution (WSE'03), 2003.

