Automated web application testing using Selenium

http://www.openqa.org/selenium/

mailto:Darius.Damalakas@gmail.com

Selenium - introduction

- Selenium tests run directly in a browser: FireFox, Internet Explorer, Mozilla.
- Browser compatibility testing.

  Test your application to see if it works correctly on different browsers and operating systems. The same script can run on any Selenium platform.
- System functional testing.

  Create regression tests to verify application functionality and user acceptance.

mailto:Darius.Damalakas@gmail.com

Platform and browser compatibility

- Windows:
  - o Internet Explorer 6.0
  - o Firefox 0.8 to 1.5
  - o Mozilla Suite 1.6+, 1.7+
  - o Seamonkey 1.0
  - o Opera 8
- Mac OS X:
  - o Safari 1.3+
  - o Firefox 0.8 to 1.5
  - o Camino 1.0a1
  - o Mozilla Suite 1.6+, 1.7+
  - o Seamonkey 1.0
- Linux:
  - o Firefox 0.8 to 1.5
  - o Mozilla Suite 1.6+, 1.7+
  - o Konqueror

mailto:Darius.Damalakas@gmail.com

How does it work?

- Selenium uses JavaScript and Iframes to embed a test automation engine in your browser. This technique should work with any JavaScript-enabled browser.

mailto:Darius.Damalakas@gmail.com

A real example

- Site - http://awesnob.com/zen/.
- Deployed Selenium - http://awesnob.com/zen/selenium/
- UI and general tests - http://awesnob.com/zen/selenium/tests/1-zen.html
- Server side and AJAX tests - http://awesnob.com/zen/selenium/tests/2-ajax.html

mailto:Darius.Damalakas@gmail.com

Selenium.Core

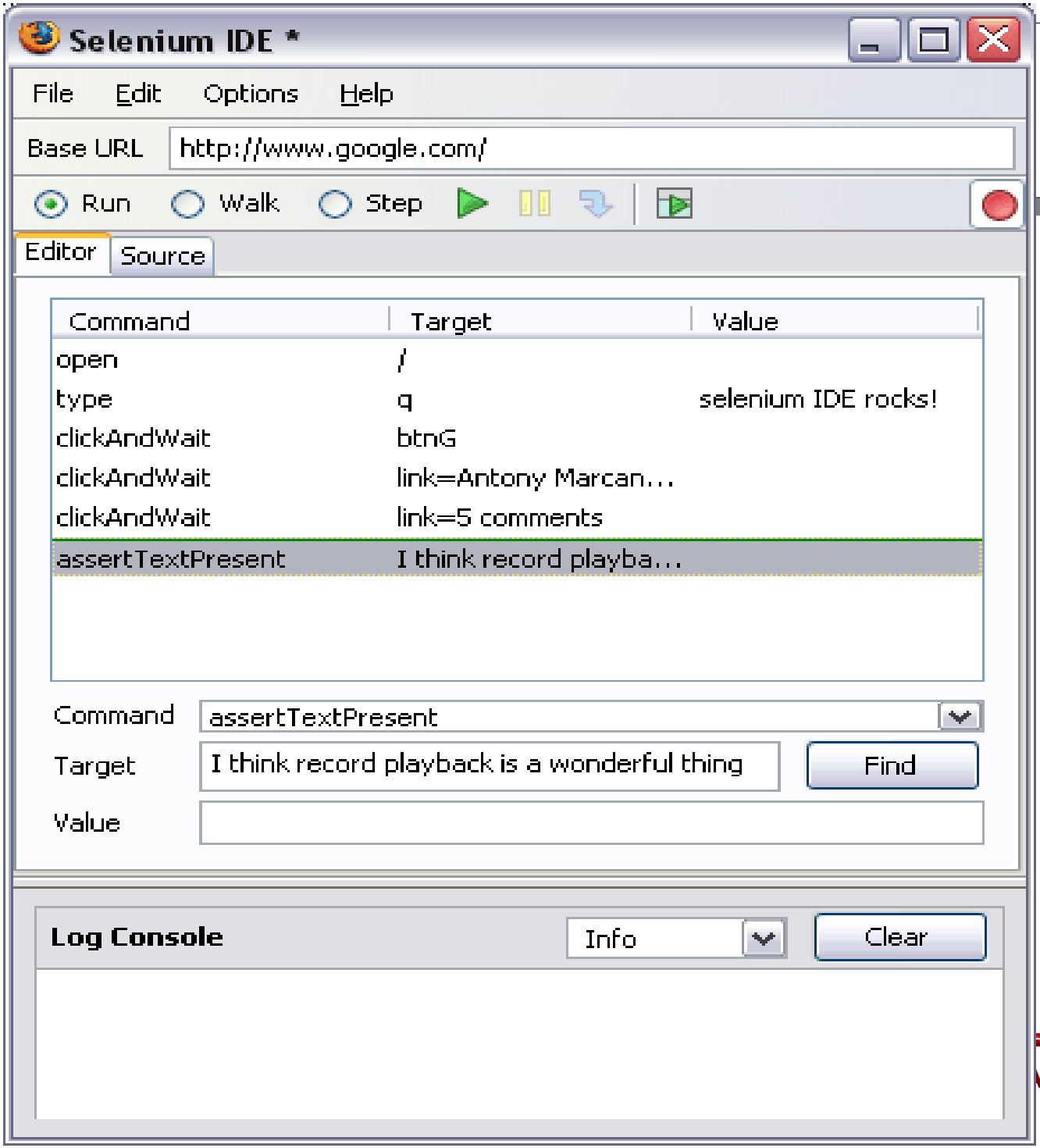
- Provides core functionality.
- Other components are built on top of Selenium.Core

mailto:Darius.Damalakas@gmail.com

Selenium.IDE

- Selenium IDE is an integrated development environment for Selenium tests. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests.
- Easy record and playback
- Intelligent field selection will use IDs, names, or XPath as needed
- Autocomplete for all common Selenium commands
- Walk through tests
- Debug and set breakpoints
- Save tests as HTML, Ruby scripts, or any other format
- Support for Selenium user-extensions.js file

Selenium.IDE example



mailto:Darius.Damalakas@gmail.com

Selenium.Remote

- Selenium Remote Control is a test tool that allows you to write automated web application UI tests in any programming language against any HTTP website using any mainstream JavaScript-enabled browser.

mailto:Darius.Damalakas@gmail.com

Selenium.Remote explained

- Selenium Remote Control provides a Selenium Server, which can automatically start/stop/control any supported browser. It works by using Selenium Core, a pure-HTML+JS library that performs automated tasks in JavaScript.
- The Selenium Server communicates directly with the browser using AJAX (XmlHttpRequest). You can send commands directly to the Server using simple HTTP GET/POST requests; that means that you can use any programming language that can make HTTP requests to automate Selenium tests on the browser. To further ease this process, we provide wrapper objects for a number of mainstream programming languages (Java, .NET, Perl, Python, and Ruby).

Selenium component comparison

	Selenium IDE	Selenium Remote Control	Selenium Core	Selenium Core HTA
Browser Support	Firefox Only	Many	All	IE Only
Requires Remote Installation	No	No	Yes	No
Supports HTTPS/SSL	Yes	Yes*	Yes	Yes
Supports Multiple Domains	Yes	Yes*	No	Yes
Requires Java	No	Yes	No	No
Saves Test Results to Disk	Yes	Yes	No	Yes

Component comparison. Legend(1)

- Browser Support: RC supports Firefox and IE out-of-the-box, and supports manual configuration of many other browsers. Core HTA runs with heightened IE security privileges
- Core HTA - Run Selenium as an "HTA" application, or "HTML Application" in Internet Explorer. HTA applications run in the security context of any trusted application on the client, so there is no cross-site scripting limitation.
- Requires Remote Installation: Selenium Core needs to be installed on the same website as the Application Under Test (AUT), because of the Same Origin Policy,
- Supports Multiple Domains: The Same Origin Policy means that tests that begin on one server/domain cannot automate applications on other servers/domains. This means that Selenium Core can't handle applications that span across multiple different domains
-

Component comparison. Legend(2)

- Requires Java: Selenium Core and Selenium IDE run directly within the browser. Selenium RC requires Java to be installed (to run the proxy server).
- Language Support: Selenium Remote Control allows you to write your tests in any programming language, including Java, .NET, Perl, Python and Ruby. (You can also add support for testing in other languages.) Selenium IDE and Selenium Core only support testing in "Selenese", a simple scripting language. Selenese has a number of strict limitations: it has no conditionals (no "if" statements), and it has no loops (no "for" statements)

Selenium licensing

- Selenium is distributed under Apache 2.0 license

- Selenium is a free (libre) software:
  - 1 – Run and use everywhere
  - 2 – Study the program and adapt to suite needs
  - 3 – Distribute for both commercial and non-commercial purposes.
  - 4 – Improve and distribute program.
- Not compatible with GPL v2.

Pros and cons – things to consider

- Selenium starts its own web server, and starts the selected browser to access it. From there Selenium can “command” the browser through javascript to invoke things on your app’s DOM.
- Every parameter is a string
- API is basically a flat bundle of methods.

- Can make use of many browsers (not restricted to IE.)
- Runs on many platforms

mailto:Darius.Damalakas@gmail.com

Other testing tools

- WatiN - http://watin.sourceforge.net/
  - Written in C#, WatiN emulates the relevant portions of browser behavior, including form submission, JavaScript,cookies and automatic page redirection, and allows C# test code to examine returned pages either as containers of forms, tables, and links.
  - When combined with a framework such as NUnit, it is fairly easy to write tests that very quickly verify the functioning of a web site.
- FitNesse - http://fitnesse.org/
  - The fully integrated standalone wiki, and acceptance testing framework
- InCisif.net - http://www.incisif.net/
  - InCisif.net is an automation tool designed to implement client-side functional testing of web applications under Internet Explorer 6.x or 7.x,
- WebAii http://artoftest.com/Products.aspx
  - WebAii™ is a web automation infrastructure that provides a rich set of features to help easily automate web applications and web scenarios.

More reading

- Comments on Selenium and Watin: http://hammett.castleproject.org/?p=120

Slide will be hosted on: http://griuvesiai.blogspot.com/2007/02/internetini-puslapi-automatizuoto.html

More info - http://griuvesiai.blogspot.com/