



Решения для автоматизации тестирования Web-приложений на базе Selenium

Быкадоров Дмитрий. ЗАО Мастерхост.
dmitry.bykadorov@gmail.com

О чем этот доклад

- **Selenium IDE:** возможности применения без использования тяжеловесных решений
- **Cubic Test:** Eclipse + GEF + Selenium = визуальное управление тестами
- **Selenium grid:** распределенная среда для тестирования web приложений – это просто
- **Вкратце о:**
 - **Selenium on Rails:** простой способ автоматизации тестирования RoR приложений
 - **Bromine:** новый проект интегрированной тестовой среды на базе Selenium

1. Selenium IDE

- **Plugin к Firefox. Позволяет:**

- Записывать тесты непосредственно из Firefox
- Воспроизводить загруженный тест в Firefox
- Воспроизводить загруженный тест в Firefox через Selenium Test Runner
- Экспортировать записанный тест в один из поддерживаемых языков (java, ruby, php, c#, python...)

- **Достоинства**

- Прост в использовании, не требует много ресурсов, не требует специальной подготовки сотрудников.
- Позволяет автоматизировать простые тестовые сценарии/операции

- **Недостатки**

- Не позволяет использовать логические условия, циклы и т.п. что ограничивает его применимость линейными тестами
- Нет возможности запуска сьюитов, а не отдельных тестов
- Нет возможности параллельного запуска (только в разных экземплярах Firefox)

Selenium IDE в действии

The screenshot shows Selenium IDE running in a Mozilla Firefox browser. The browser window displays the Google search page. The Selenium IDE interface is overlaid on the page, showing a table of test commands:

Command	Target	Value
open	http://www.google.ru/	
clickAndWait	//img[@alt="Invention of the ..."]	
click	link=Laser - Wikipedia, the fr...	
click		
clickAndWait	link=Расширенный поиск	

The browser shows the Google search results for "first laser".

The screenshot shows Selenium IDE running in a Mozilla Firefox browser. The browser window displays a page titled "Selenium Functional Test Runner v0.8.2 [1.22.7]". The Selenium IDE interface is overlaid on the page, showing a table of test commands:

Command	Target	Value
clickAndWait	link=Расширенный поиск	
selectWindow	null	
select	as_epq	laser
type	as_oq	first
clickAndWait	btnG	
clickAndWait	link=Настройки	
click	nof	
click	stf	
clickAndWait	submit2	
assertAlert	Настройки сохранены.	
verifyTextPresent	First Laser	

The Selenium TestRunner window is also visible, showing the test suite and its execution status:

Tests	Commands
1 run	0 passed
1 failed	0 failed
	1 incomplete

The browser shows the search results for "first laser first 'laser'".

The screenshot shows the Selenium IDE application window. The "Export Test As..." menu is open, displaying a list of export options:

- HTML
- Java - Selenium RC
- C# - Selenium RC
- Perl - Selenium RC
- PHP - Selenium RC
- Python - Selenium RC
- Ruby - Selenium RC
- Ruby - Selenium IDE 0.7 (deprecated)

The Selenium IDE interface also shows the "Command" field with a dropdown menu, the "Target" field, and the "Value" field. The "Log" and "Reference" tabs are visible at the bottom.

2. Cubic Test

- **Возможности:**

- Интегрируется в Eclipse IDE как отдельная Perspective.
- Имеет инструменты Record/Playback.
- Использует визуальное моделирование и управление тестами (на базе Graphical Test Editor, GEF, также интегрируемого в Eclipse).
- Позволяет выносить общие сценарии в субтесты и подключать их по мере необходимости.
- Позволяет объединять тесты в наборы (сюиты), также используя визуальное представление.
- Позволяет экспортировать графическое представление тестов в HTML Prototype или Watir (в том числе допускает написание собственных экспортеров).

- **Достоинства**

- Оригинальная и простая для понимания концепция визуального управления тестами, основанная на распространенных и доступных open source инструментах.
- Прост в использовании и не требует специализированных навыков программирования на том или ином языке (java, ruby, c#...).
- Встроенные средства записи и воспроизведения.

- **Недостатки**

- Отсутствие возможности параллельного воспроизведения тестовых наборов.
- Некорректная работа с кириллицей

Cubic Test в действии

The image displays the Cubic Test Eclipse IDE interface, showing a test flow for the OZON.ru website and a detailed view of a test step.

Left Window (basics): Shows a test flow starting with the URL `http://www.ozon.ru/`. The flow includes steps for finding search text, entering text, clicking 'Найти', navigating to the next page, and clicking 'С этими товаром часто покупают'.

Right Window (probe): Shows a detailed view of a test step for the URL `http://shop.mh1.masterhost.ru`. The step involves clicking a button labeled 'beginner' and then 'd1'.

Properties Window: Shows the properties for the selected test step, including the Name `[.m] masterhost: [.m]??????` and the Description.

3. Selenium grid

- **Возможности:**

- Быстрое и простое распараллеливание выполнения тестов. В основе данной возможности лежит фреймворк TestNG (а не junit как у «классических» selenium-тестов).
- Возможность построения распределенной и масштабируемой среды для выполнения тестов.

- **Достоинства**

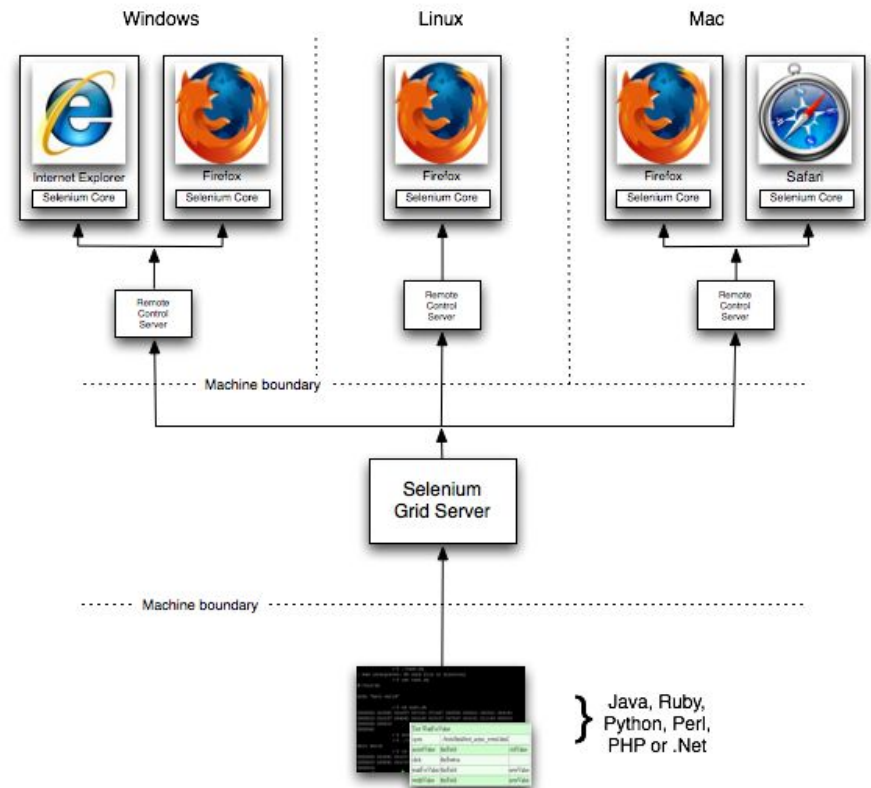
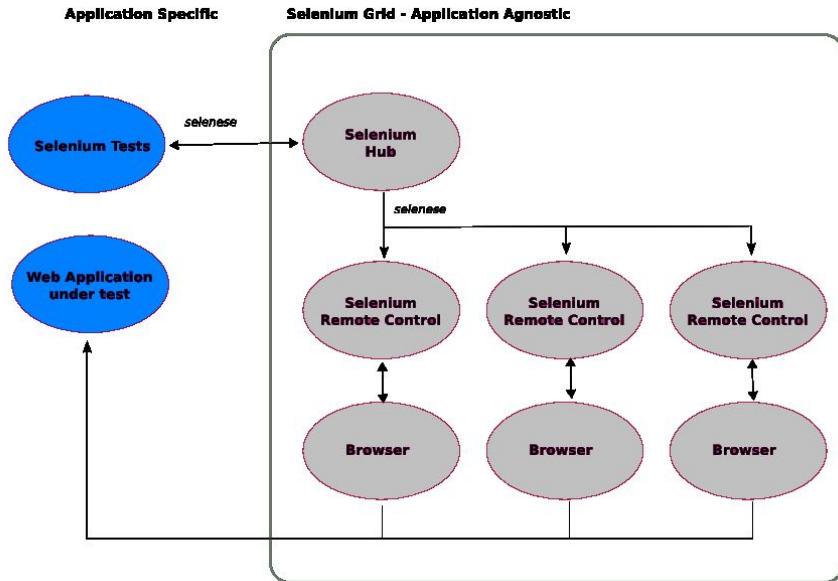
- Многократное уменьшение времени выполнения при большом количестве тестовых сценариев.
- Возможность использования ранее написанных тестов (на java, ruby python...).
- Простой способ построения распределенной среды для выполнения тестов.

- **Недостатки**

- Нет средств Record/Playback.
- Нет поддержки selence test cases.
- Требуется более высокой квалификации от сотрудников.
- Сыроват. Например:
 - Проблемы с кириллицей при воспроизведении.
 - Проблемы с запуском parallels tests.

Архитектура Selenium-grid

Selenium Grid Setup



Selenium-grid в действии

```
public class TestGridDemo
{
    @BeforeTest(alwaysRun = true)
    @Parameters({"seleniumHost", "seleniumPort", "browser", "webSite"})
    protected void startSession(String seleniumHost, int seleniumPort, String browser, String webSite)
    throws Exception {
        startSeleniumSession(seleniumHost, seleniumPort, browser, webSite);
    }
    @AfterTest(alwaysRun = true)
    protected void closeSession() throws Exception {
        closeSeleniumSession();
    }
    @Test(enabled = true, groups = {"cp", "registration"}, description = "Grid test demo")
    @Parameters({"seleniumHost", "seleniumPort", "browser", "webSite"})
    public void SomeTest() throws Exception
    {
        session().open("http://domain.com");
    }
}
```

4. О чем еще стоит упомянуть?

- Selenium on Rails. Простой способ интеграции Selenium с Ruby on Rail приложениями.
 - Selenium Core не «замусоривает» /public.
 - Не нужно создавать suite-файлы, они генерятся на лету — один suite на директорию в /test/selenium.
 - Возможность использования нескольких более удобных форматов для TestCases, помимо HTML
 - Selenese, самый простой тестовый формат.
 - Rselenese, позволяет писать тесты на ruby.
 - HTML/RHTML.
 - Partial test cases. Позволяет выносить часто используемые последовательности действий в отдельные «кусочные» кейсы и подключать их по мере надобности.
 - Загрузка фикстур и имитация сессий.

4. О чем еще стоит упомянуть?

- Bromine. Интегрированная тестовая среда на базе Selenium.
 - Selenium предоставляет возможности
 - Создание тестов при помощи IDE
 - Предоставляет JS framework
 - Предоставляет Remote Control server
 - Предоставляет Core runner
 - Bromine, возможности:
 - Многофункциональный QA инструмент
 - Позволяет создавать проекты
 - Привязывать требования к проектам
 - Привязывать тесты к требованиям
 - Предоставляет простой способ управления и запуска тестов
 - Позволяет анализировать результаты запуска тестов
 - Позволяет создавать дефекты
 - Также имеется облегченная light версия только для запуска тестов и анализа результатов

Вопросы?

Буду рад ответить на ваши вопросы:

- сейчас
- в кулуарах
- по email
 - dmitry.bykadorov@gmail.com
 - dbykadorov@masterhost.ru