Testing Web Applications with Scripting Languages

Mark Rees
Century Software

twill

- simple language for web browsing
- stress tester
- web site tester
- http://twill.idyll.org

twill-sh

- interactive shell
  - twill-sh
- scripted browsing
  - twill-sh -u http://localhost:5000 test-it
- record twill scripts with scotch
  - http://darcs.idyll.org/~t/projects/scotch/

code

twill-fork

- scripted stress testing
  - twill-fork -n 100 -p 5 test-wiki.twill

code

twill-unit testing

- twill-sh process directory of tests
- twill master run script
- python unit test framework

code

twill-other features

- supports authentication
- debug helpers
- extended with user defined commands
- handles "bad" html
- "in-process" wsgi support

twill-cons

- advanced usage requires python knowledge
- it's knows nothing about javascript

selenium

- test web applications from within the browser
- linux, mac os x, windows
  - ie6+, firefox 0.8-2.0, mozilla 1.6+, seamonkey 1.0, opera 8, camino 1.0a1, konqueror
- http://www.openqa.org/selenium

selenium core

- uses javascript & iframes to embed test engine
- core must be installed on webserver under test

selenium ide

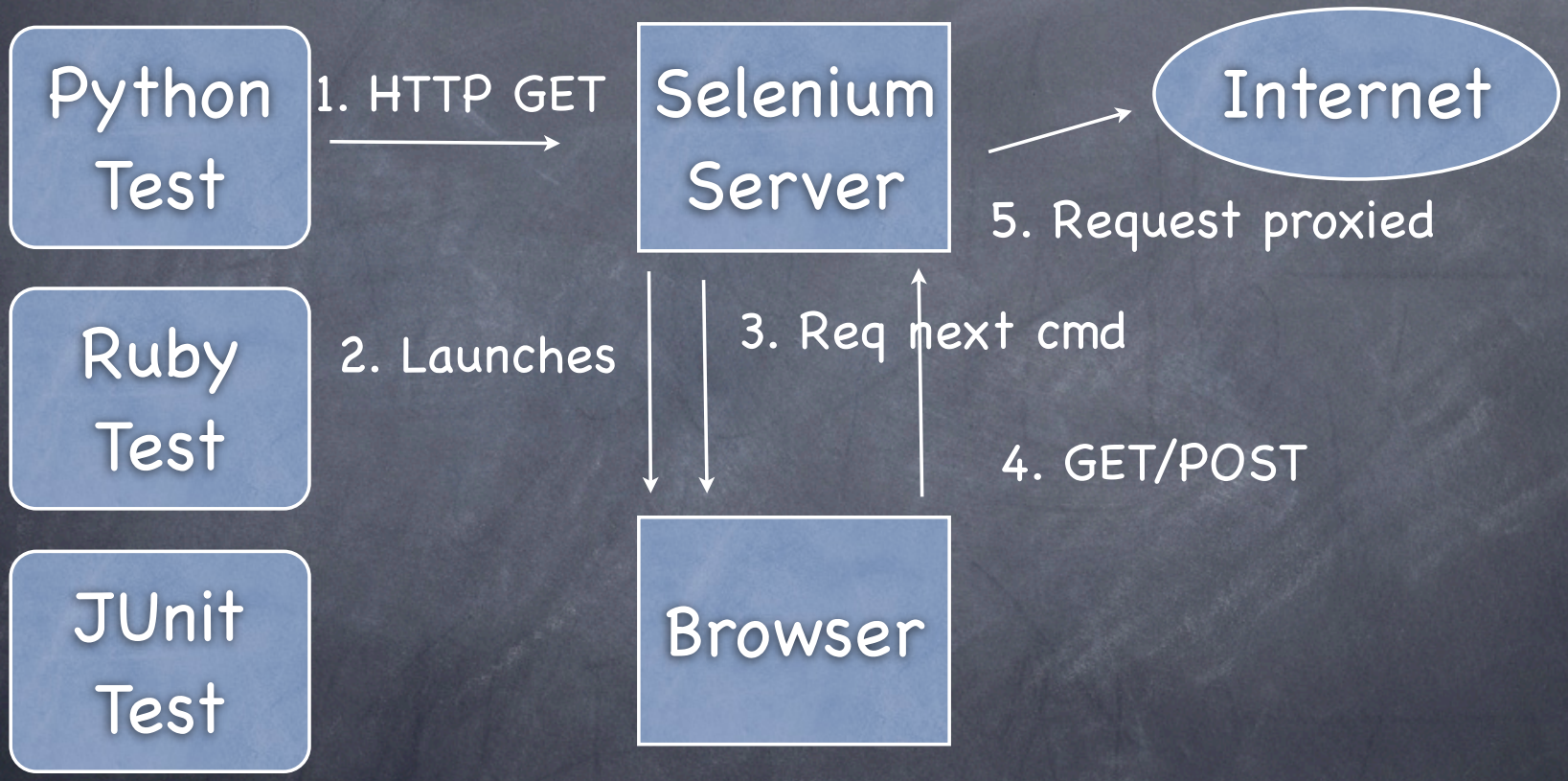
- firefox extension - ide for selenium tests
- gets round core limitations
- saves tests as html or ruby scripts

selenium remote control

- allows auto web ui testing with any language
- 2 components
  - selenium server, written in java
  - drivers for mainstream languages including python and ruby

code

selenium: how it works



other options

- Watir http://www.openqa.org/watir
- Watij http://www.openqa.org/watij
- Floyd http://www.openqa.org/floyd
- HttpUnit http://httpunit.sf.net

the missing paper

- http://hex-dump.googlecode.com/trunk/osdc/2007/testing-web-apps.pdf

questions

code slides follow

code:test-wiki.twill

```
go http://localhost:5000
follow "Edit FrontPage"
fv 1 content "Welcome to the QuickWiki front page
\nWhatIsTwill"
submit
follow WhatIsTwill
follow "Create the page"
fv 1 content "Twill is an interactive or scripted web
emulator, useful for web site inspection, unit,
functional and stress testing."
submit
```

back

code:stress-wiki.twill

```
go http://localhost:5000
follow "Title List"
follow "FrontPage"
```

back

code: twill_unit_test_wiki.py

```
class TestWiki(unittest.TestCase):

    wikiHost = 'localhost'
    wikiPort = str(5000)
    wikiUrl = "http://"+wikiHost + ":" + wikiPort

    def setUp(self):
        startWebServer()

    def testEditPage(self):
        twill.execute_file("test-edit-page.twill",
                           initial_url=self.wikiUrl)

    def tearDown(self):
        stopWebServer()
```

back code

code:test-edit-page.twill

```
follow "Edit FrontPage"
fv 1 content "Welcome to the QuickWiki front page.
WhatIsTwill"
submit
find "WhatIsTwill"
```

next

code: selenium_test_wiki.py

```
    def testDeletePage(self):
        selenium = self.selenium
        selenium.open("/")
        selenium.click("link=Title List")
        selenium.wait_for_page_to_load(5000)
        selenium.capture_screenshot("delete_page_before.png")
        selenium.drag_and_drop_to_object("page-TestPage",
"trash")
        selenium.capture_screenshot("delete_page_after.png")
        self.failIf(selenium.is_element_present("page-
TestPage"))
```

back