

Flash Talk: Geb

Tomasz Kalkosiński

@tkalkosinski



Warszawa, 9.04.2013 r.



Geb - wprowadzenie



Geb is a browser automation solution.

www.gebish.org

- język Groovy
- sterownik WebSriver (Selenium 2)
- wbudowany wzorzec Page Object Model
- składnia wzorowana na jQuery
- frameworki Spock, JUnit, TestNG
- The Book Of Geb

Geb - możliwości

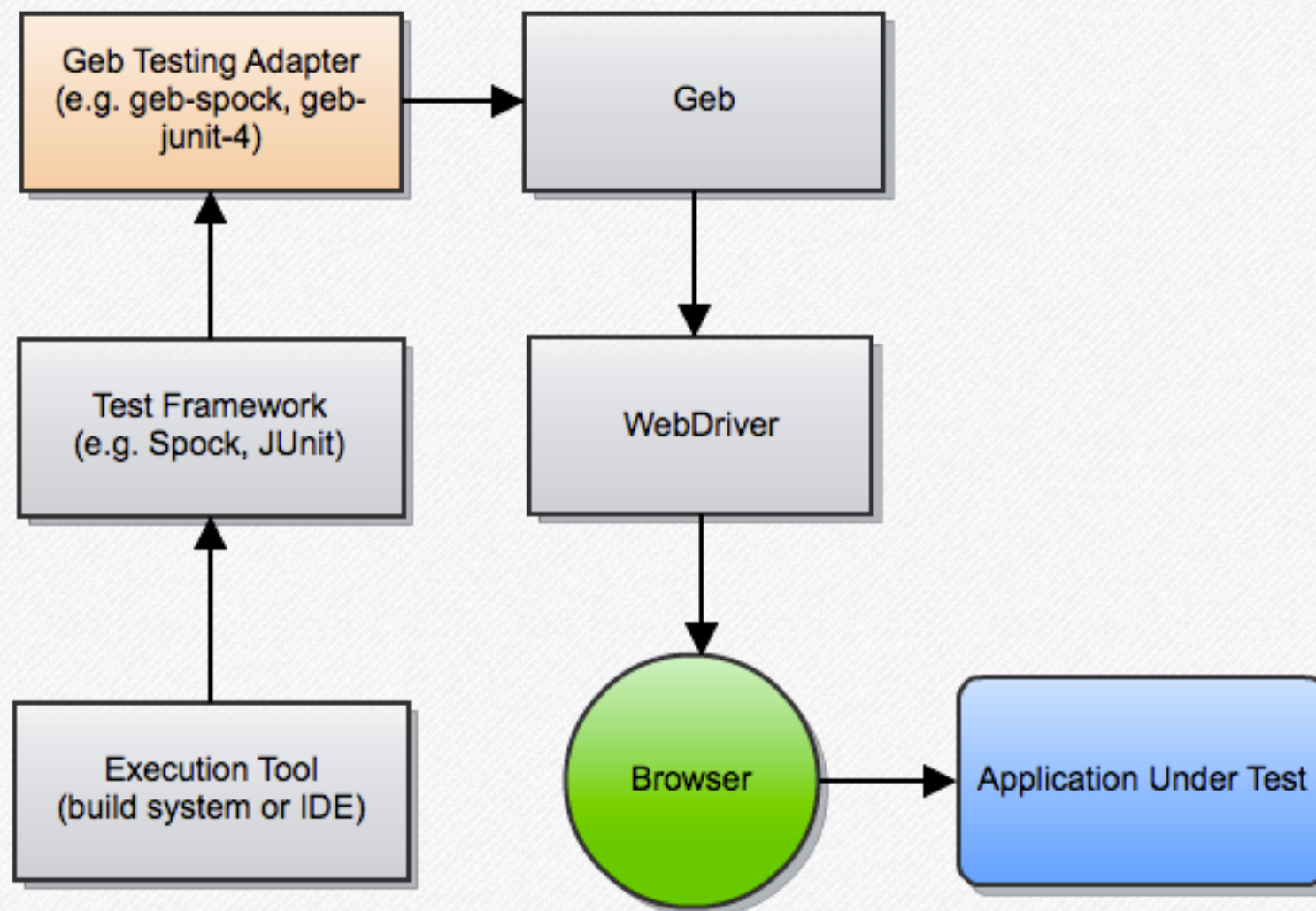


It uses Groovy's dynamism to remove the noise and boiler plate code.

The Book Of Geb

- skrypty lub specyfikacje
- utrzymywanie sesji
- wsparcie dla asynchroniczności
- obsługa CSS i JavaScript
- pobieranie plików ze stron
- wsparcie dla wielu przeglądarek

Architektura Geba



<http://danspeck.blogspot.com/2012/10/springone-2gx-conference-summary.html>

Przykład

```
import geb.Browser

Browser.drive {
  go "http://myapp.com/login"

  assert $("h1").text() == "Please Login"

  $("form.login").with {
    username = "admin"
    password = "password"
    login().click()
  }

  assert $("h1").text() == "Admin Section"
}
```

Wyszukiwanie

```
// elementy według składni CSS oraz indeksu  
$("div")  
$("div.main", 0)
```

```
// wyszukiwanie po atrybutach  
$("h1", 2, class: "heading")  
$("p", text: contains("Geb"))  
$("input", value: ~/\d{3,}-\d{3,}-\d{3,}/)
```

```
// składnia podobna do jQuery - łańcuchy  
$("div").find(".b")  
$("div").filter(".c").parents()  
$("p.c").siblings()
```


Page Object Model

```
import geb.Page

class LoginPage extends Page {
    static url = "http://myapp.com/login"
}
```

```
import geb.Browser

Browser.drive {
    to LoginPage
}
```

Page Object Model

```
import geb.Page

class LoginPage extends Page {
  static url = "http://myapp.com/login"
  static at = { title == "Please Login" }
}
```

```
import geb.Browser

Browser.drive {
  to LoginPage
  assert at(LoginPage)
}
```


Page Object Model

```
import geb.Page

class LoginPage extends Page {
  static url = "http://myapp.com/login"
  static at = { title == "Please Login" }
  static content = {
    username { $("#username") }
    password { $("#password") }
    login { $("#button", type: "submit") }
  }
}
```


Page Object Model

```
import geb.Browser

Browser.drive {
  to LoginPage
  assert at(LoginPage)
  username = "admin"
  password = "password"
  login.click()
  assert at(AdminPage)
}
```


Połączenie ze Spockiem

- klasa testowa `GebReportingSpec` - zapisywanie stanu przeglądarki po każdym kroku

```
reports/geb/my/tests/FunctionalSpec/1-1-login-login  
screen.html  
reports/geb/my/tests/FunctionalSpec/1-1-login-login  
screen.png  
reports/geb/my/tests/FunctionalSpec/1-2-login-end.html  
reports/geb/my/tests/FunctionalSpec/1-2-login-end.png
```

- adnotacja `@Stepwise`
 - Zachowana kolejność kroków wewnątrz specyfikacji
 - Utrzymywanie sesji (cookies) wewnątrz specyfikacji

Przykładowa specyfikacja

```
@Stepwise
class AdminLoginSpec extends GebReportingSpec {

  def 'should log in as admin'() {
    when:
      go LoginPage
      username = "admin" ; password = "pass" ;
login.click()
    then:
      at WelcomePage
      heading.text() == "Welcome admin!"
  }

  def 'should log out as admin'() {
    when:
      logout.click()
    then:
      at LoginPage
      heading.text() == "You have been logged out."
  }
}
```


Live

Nie dołączam przykładowego kodu.

Gotowe przykłady można znaleźć tutaj: <https://github.com/geb>

Pytania?

Dziękuję

Tomasz Kalkosiński

@tkalkosinski

tkl@touk.pl

<http://refaktor.blogspot.com>



Warszawa, 9.04.2013 r.

