



RichFaces – Testing on Mobile Devices

Pavol Pitoňák

Quality Assurance Engineer, Red Hat

17 Feb 2012

Why Are We Doing This?

- canalys.com



487.7 mil



414.6 mil



RichFaces

Maven

Arquillian

TestNG

Jenkins

Android

Selenium

Eclipse

JBoss AS



Testing a Java Web Application

1. Build archive
2. Start a container
3. Deploy archive
4. Open web browser
5. Run tests (continuously)
6. Review test results



RichRates

- github.com/qa/richrates
- richrates-ppitonak.rhcloud.com

The screenshot displays the RichRates website interface. At the top right, there is a dropdown menu labeled "Emerald Town". Below this is the "RichRates" logo in green. A navigation bar contains three tabs: "Calculator", "Exchange Rates", and "Chart". The main content area features a calculator for converting EUR to CZK. It shows the European Union flag and the Czech Republic flag with a double-headed arrow between them. The "Date" field is set to "Jan 11, 2012" and the "Amount" field is set to "45.0". A green "Calculate" button is positioned below the amount field. The result is displayed as "45.000 EUR = 1,161.945 CZK". To the right of the calculator is a grid of 30 national flags. At the bottom of the page, a footer reads: "Powered by RichFaces 4. Data provided by European Central Bank. Flags by RedpixArt."



Ruby ▾

RichRates













Calculator • Exchange Rates • Chart

Daily Exchange Rates

These reference exchange rates are being published by European Central Bank, usually at 14:15 CET on days when the payment system TARGET is open.

Date:

Wednesday, March 3, 2010

Currency ▾	Currency Name ▾	Exchange Rate ▾
 CNY	Chinese Yuan Renminbi	9.3111
 JPY	Japan Yen	121.1
 LVL	Latvian Lats	0.7093
 CZK	Czech Koruna	25.765
 RON	Romanian Lei	4.1005
 MXN	Mexican Peso	17.3653
 CAD	Canadian Dollar	1.4097
 ZAR	South African Rand	10.2804
 AUD	Australian Dollar	1.5116
 NZD	New Zealand Dollar	1.9767
 GBP	British Pound	0.9057
 NOK	Norwegian Kroner	8.0745
Currency	Currency Name	Exchange Rate

[Previous](#) **1** **2** **3** [Next](#)

Powered by RichFaces. Data provided by European Central Bank. Flags by RedpixArt.



richrates-ppitonak.rhcloud.com



richrates-ppitonak.rhcloud.com

Calculator Table Chart

Date: 

Amount:


Calculate








richrates-ppitonak.rhcloud.com

Calculator Table Chart

These reference exchange rates are being published by European Central Bank, usually at 14:15 CET on days when the payment system TARGET is open.

Date: 

Thursday, February 16, 2012

Currency	Exchange Rate
 Australian Dollar (AUD)	1.2168
 Bulgarian Lev (BGN)	1.9558
 Brazilian Real (BRL)	2.2552
 Canadian Dollar (CAD)	1.3026
 Swiss Franc (CHF)	1.2072



richrates-ppitonak.rhcloud.com



Project Structure

- Standard Maven structure
- testng.xml
- testng-ftest.xml
- testng-all.xml



Running Unit Tests

`mvn clean package`

`testng.xml`



Running Functional Tests

```
mvn clean package -P jbossas-managed-71,ftest
```

```
testng-ftest.xml
```



Running All Tests

```
mvn clean package -P jbossas-managed-71,all-tests
```

```
testng-all.xml
```



Sample Unit Test

```
public class UAgentInfoTest {  
    @Test  
    public void testIPad() {  
        Assert.assertTrue(...);  
    }  
}
```



Sample Arquillian Test

```
public class SkinBeanTest extends Arquillian {  
    @Inject private SkinBean skinBean;  
    @Deployment  
    public static JavaArchive createTestArchive() {  
        return ...;  
    }  
    @Test  
    public void testDefaultSkin() {  
        Assert.assertEquals(skinBean.getSkin(), "blueSky");  
    }  
}
```



Sample WebDriver Test

```
public class SampleTest extends Arquillian {  
    @Drone public WebDriver driver;  
    @ArquillianResource private URL deployedRoot;  
    @Deployment  
    public static WebArchive createTestArchive() {  
        return ...;  
    }  
    @Test  
    public void testDefaultSkin() {  
        driver.get("http://devconf.cz");  
        ...  
    }  
}
```



Deployment

```
@Deployment(testable = false)
public static WebArchive createTestArchive() {
    WebArchive war =
    ShrinkWrap.createFromZipFile(WebArchive.class, new
    File("target/richrates.war"));
}
```

... or construct the WAR yourself :)



Deployment

```
@Deployment(testable = false)
public static WebArchive createTestArchive() {
    WebArchive war =
    ShrinkWrap.createFromZipFile(WebArchive.class, new
    File("target/richrates.war"));
}
```

... or construct the WAR yourself :)



Deployment 2

```
@Deployment(testable = false)

public static WebArchive createTestArchive() {

    MavenDependencyResolver resolver = DependencyResolvers.use(MavenDependencyResolver.class).loadMetadataFromPom("pom.xml");

    WebArchive war = ShrinkWrap.create(WebArchive.class, "richrates.war");

    war.addPackage("org.richfaces.examples.richrates");

    war.addPackage("org.richfaces.examples.richrates.annotation");

    war.addPackages(true, "org.slf4j");

    war.merge(ShrinkWrap.create(ExplodedImporter.class, "tmp1.war").importDirectory("src/main/webapp").as(WebArchive.class));

    war.merge(ShrinkWrap.create(ExplodedImporter.class, "tmp2.war").importDirectory("src/main/resources").as(WebArchive.class), "WEB-INF/classes");

    war.addAsLibraries(resolver.artifact("org.richfaces.ui:richfaces-components-ui").resolveAsFiles());

    war.addAsLibraries(resolver.artifact("org.richfaces.core:richfaces-core-impl").resolveAsFiles());

    war.addAsLibraries(resolver.artifact("org.jdom:jdom").resolveAsFiles());

    war.addAsLibraries(resolver.artifact("joda-time:joda-time").resolveAsFiles());

    return war;

}
```



Arquillian.xml

```
<extension qualifier="webdriver">  
  <property  
name="implementationClass">org.openqa.selenium.firefox.FirefoxDriver</property>  
</extension>
```

-Darquillian.browser.implementation.class=...



Arquillian.xml

- Define your containers
- Maven profiles



Sample WebDriver Test

```
public class SampleTest extends Arquillian {  
    @Drone public WebDriver driver;  
    @ArquillianResource private URL deployedRoot;  
    @Deployment  
    public static WebArchive createTestArchive() {  
        return ...;  
    }  
    @Test  
    public void testDefaultSkin() {  
        driver.get("http://devconf.cz");  
        ...  
    }  
}
```



Page Factory

```
@Test
```

```
public void googleSearchTest {
```

```
    WebDriver driver = ...
```

```
    driver.get("http://www.google.com/");
```

```
    GoogleSearchPage page =  
    PageFactory.initElements(driver, GoogleSearchPage.cl  
    ass);
```

```
    page.searchFor("Cheese");
```

```
}
```



Page Factory

Calculator Table Chart

Date: 

Amount:



Page Object

```
public class CalculatorPage extends AbstractPage {  
    @FindBy(id = "calculator:amount")  
    private WebElement amountInput;  
    @FindBy(xpath = "//div[contains(@class, 'result')]")  
    private WebElement resultOutput;  
    @FindBy(id = "calculator:calculateButton")  
    private WebElement submitButton;  
    ...  
}
```



Mobile Web Testing

- Almost no changes in Arquillian config
- Android 1.x not supported
- Android 2.x – phones
- Android 3.x – tablets
- Android 4.x – phones and tablets

- Physical device vs. emulator



Physical Devices

- Windows – install USB drivers
- Linux – set up udev
- Mac OS X – no setup needed



Android Emulator

- Part of Android Software Development Kit
- Android 1.5 – 4.0.3

- `<sdk_path>/tools/android`
- Eclipse plugins



Emulators

- 2.2
 - Relatively fast
 - Works with selenium-server 2.6
- 2.3
 - Unreliable
- 3.x
 - Slow
- 4.x
 - Relatively fast
 - Works well with newest Selenium



Demo

Creating and launching a new emulator



Android-x86 Project

- www.android-x86.org
- Project porting Android to x86 platform



Demo

Launching Android in VirtualBox



AndroidDriver Config

```
<extension qualifier="webdriver">
```

```
<property name="implementationClass">  
org.openqa.selenium.android.AndroidDriver  
</property>
```

```
<property name="remoteAddress">
```

```
http://localhost:4444/wd/hub
```

```
</property>
```

```
</extension>
```



Emulator Configuration

- Install android-server.apk
- Launch WebDriver application in emulator
- `platform-tools/adb forward tcp:4444 tcp:8080`
- Use 10.0.2.2 for host's localhost



VirtualBox Configuration

- Set up network
- `platform-tools/adb connect <android_ip>`
- Install `android-server.apk`
- Launch WebDriver application in emulator
- `platform-tools/adb forward tcp:4444 tcp:8080`
- Tests need server IP address

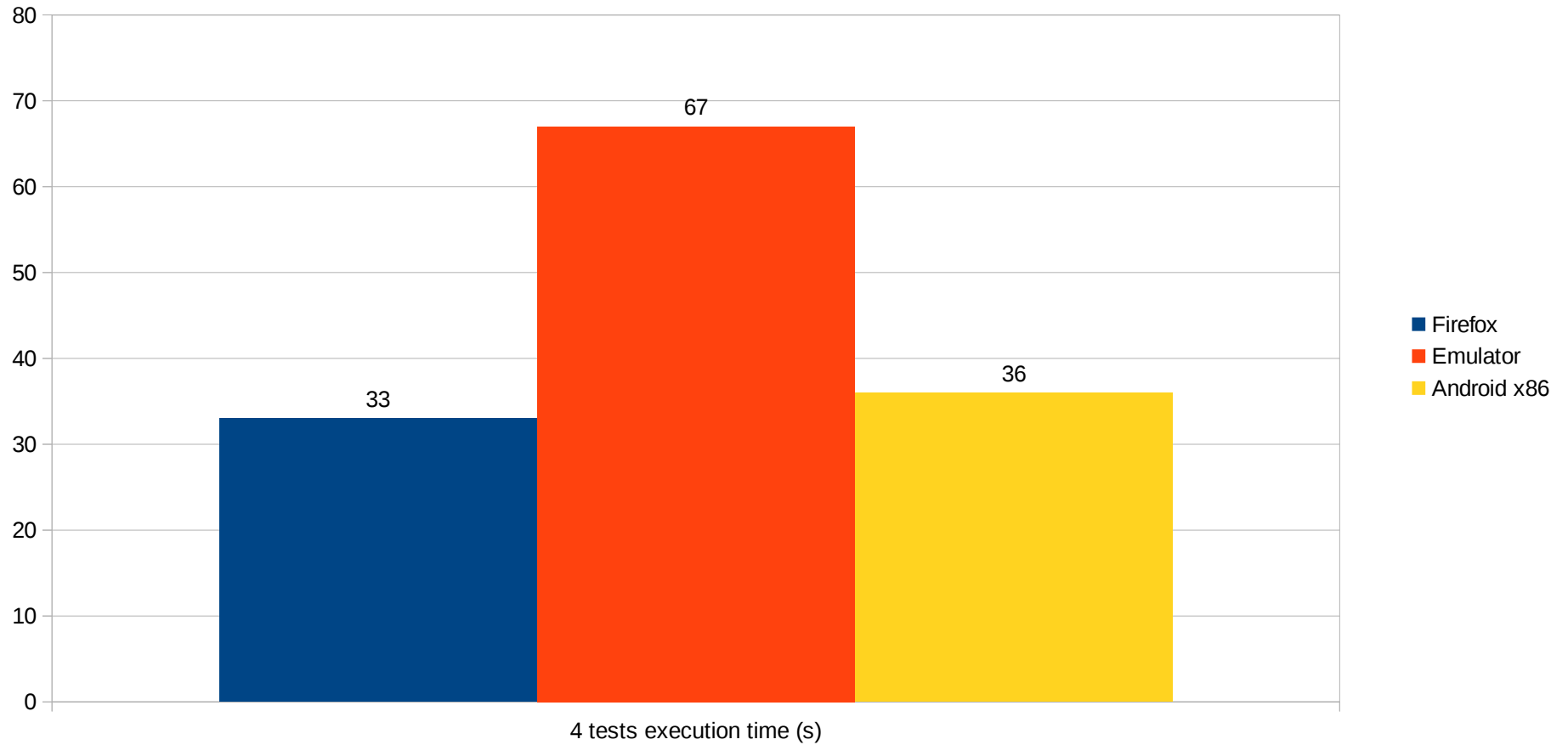


Demo

Tests running with Firefox, Android emulator and
Android-x86



Fast, Faster, Fastest



Eclipse

- Plugins
 - Android
 - JBoss Tools
 - m2e
 - TestNG

- Supports both emulator and Android-x86



Demo

Running tests in Eclipse



Debugging

- You already know how :)



Demo

Debugging



Arquillian Android Extension

- github.com/kpiwko/arquillian-extension-android
- Hot news!

- Create emulator
- Start emulator
- Install Android server
- Start Android server
- Can connect to real device



Console Logging Listener

- Usual console log:



Console Logging Listener

- Our console log:

[10:47:54] STARTED: ftest.ExchangeRatesTest.testInit()

[10:47:54] SUCCESS: ftest.ExchangeRatesTest.testInit()

[10:47:55] STARTED: ftest.ExchangeRatesTest.testNextAndPrevious()

[10:47:57] SUCCESS: ftest.ExchangeRatesTest.testNextAndPrevious()

[10:48:09] STARTED: ftest.CalculatorTest.testCalculate()

[10:48:12] SUCCESS: ftest.CalculatorTest.testCalculate()



Console Logging Listener

- Register in pom.xml or in test class

```
@Listeners(ConsoleStatusTestListener.class)
```

```
public abstract class AbstractWebDriverTest<P  
extends Page> extends Arquillian {
```

```
    ...
```

```
}
```




Screenshot on Failure

```
@AfterMethod(alwaysRun = true, groups = "arquillian")
public void handleTestError(ITestResult result) {
    if (result.getStatus() == ITestResult.SUCCESS) {
        return;
    }
    ...

    File scrFile = ((TakesScreenshot)
driver).getScreenshotAs(OutputType.FILE);
    FileUtils.copyFile(scrFile, imageOutputFile);
}
```





Jenkins Android Plugin


Run an Android emulator during build 


Run existing emulator

Run emulator with properties

Android OS version 

Screen density 

Screen resolution 

Device locale 

- en_AU
- en_CA
- en_GB**
- en_NZ
- en_SG
- en_US



Jenkins Android Plugin

Run an Android emulator during build 

Run existing emulator 

AVD name

Enter the name of an existing Android emulator configuration



Run emulator with properties 

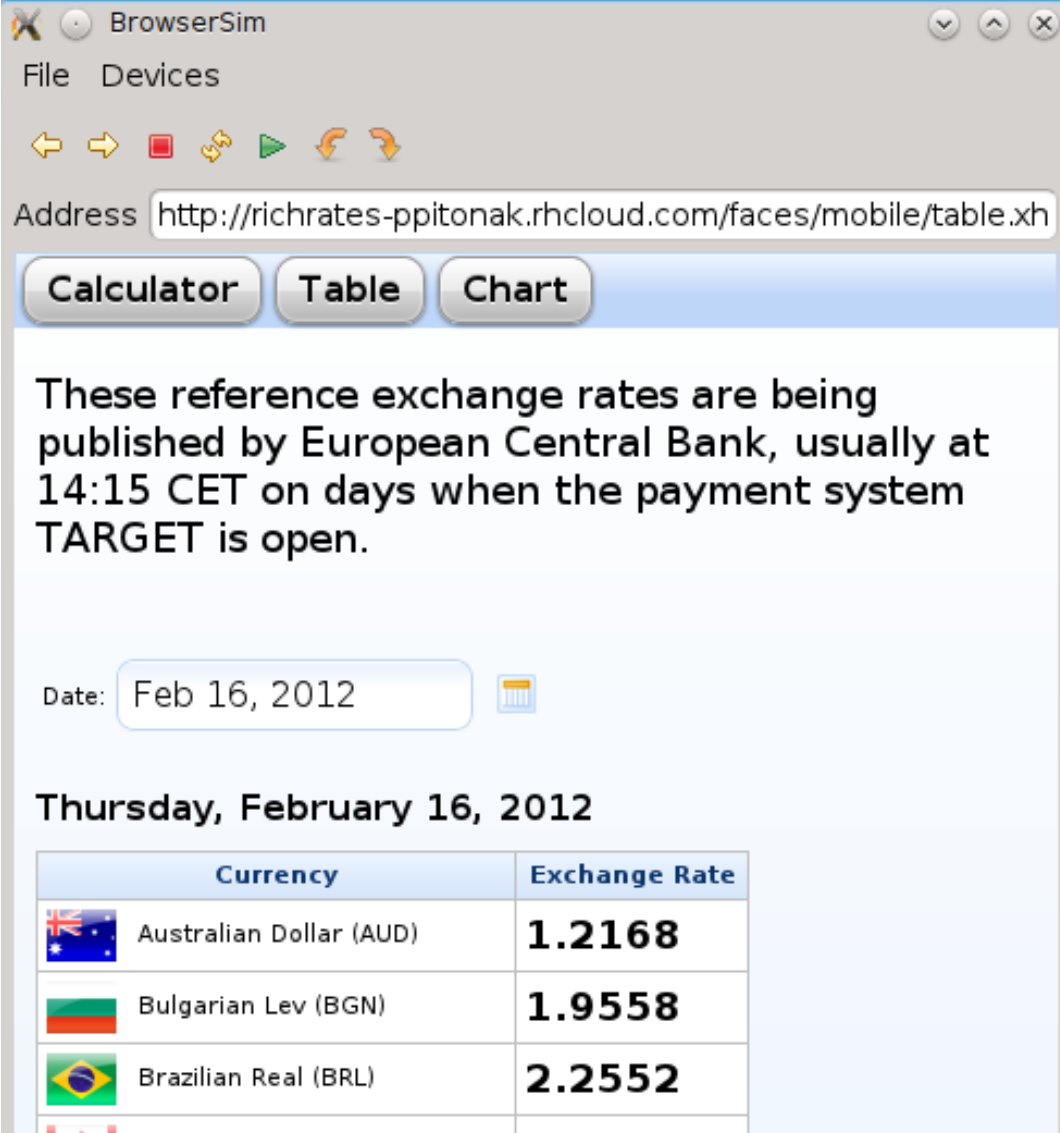


Mobile Browser Simulator




- JBoss Tools 3.3 M5
- Simulates mobile phones and tablets
- Perfect for layout testing
- Not for automation



Mobile Browser Simulator



The screenshot shows a mobile browser simulator window titled "BrowserSim". The address bar contains the URL "http://richrates-ppitonak.rhcloud.com/faces/mobile/table.xh". Below the address bar are three buttons: "Calculator", "Table", and "Chart". The main content area displays a disclaimer: "These reference exchange rates are being published by European Central Bank, usually at 14:15 CET on days when the payment system TARGET is open." Below this is a date input field set to "Feb 16, 2012" with a calendar icon. The date is confirmed as "Thursday, February 16, 2012". A table of exchange rates is shown below:

Currency	Exchange Rate
 Australian Dollar (AUD)	1.2168
 Bulgarian Lev (BGN)	1.9558
 Brazilian Real (BRL)	2.2552



Conclusion

- Arquillian + Selenium 2
- Run from console and Eclipse
- How to create Android emulator
- Android-x86
- Emulator snapshots
- ID locators, Xpath locators, CSS locators



ppitonak@redhat.com
@ppitonak

jboss.org/richfaces

Q/A

