

Outils d'automatisation de tests d'interfaces Web

David GERBAULT

Ingénieurs 2000
Xposé 2010-2011

1

WEBOGRAPHIE

<http://seleniumhq.org/>

<http://www.clever-age.com/>

<http://watir.com/>

Pourquoi ce sujet?

PLAN

1. Présentation
2. Selenium
3. Watir
4. Conclusion

CONSTAT !

Souvent : tests du Contrôleur (code)

Rarement : tests de la Vue (Interface Web)

OBJECTIF

Simuler l'interaction de(s) l'utilisateur(s)
au travers de tests fonctionnels

LES TESTS...

Automatisation de tests

Selenium / CubicTest / Watir / AutoIT / Canoo / WebTest

~~Gestion des tests~~

Selenium / Outils borland / Salome / TestLink / Fitnessse / Outil HP / Twist

~~Tests de charge~~

Grinder / OpenSta / JMeter

~~Model Checking~~

NModel / Smart testing / MaTeLo

Qu'est ce que c'est ? : Vérification algorithmique d'un modèle donné, du système lui-même ou d'une abstraction du système, pour savoir si il satisfait une spécification, souvent formulée en termes de logique temporelle.

= => Curieux? **Google it !**

QUI ?

Analystes programmeur

Conformité des développements par rapport aux besoins exprimés

Développeurs et équipes d'assurance qualité

Valider le bon fonctionnement de l'application et le passage en production.

QUAND ?

Le test doit être joué plus d'une fois.

&

Le test peut être automatisé de bout en bout.

COMMENT?

1. Lecture des besoins & spécifications.
2. Définition du périmètre de test.
3. Rédaction des cas de tests.
4. Enregistrement des tests dans l'outil.
5. Exécution des tests.

CONSÉQUENCES?

- Tests de non régressions fréquents
- Retour rapide aux développeurs pendant la phase de développement
- Cas de tests « illimités »
- Rapports personnalisés
- Convient aux développements Agile et d'eXtreme Programming
- Documentation rigoureuse des cas de tests
- Suppression des erreurs des tests manuels

INTÉGRATION CONTINUE

Ecrire des tests dans le format voulu et adapté à l'outil d'intégration continue.

Exemple

Selenium – Java (JUnit) – Eclipse – Hudson

Exemple avec ant + Selenium + Hudson

<http://www.clever-age.com/veille/blog/integration-pas-a-pas-de-tests-selenium-dans-hudson.html>

PROBLÈMATIQUES

Ce qu'il faut tester

- Contenus
- Liens
- Fonctions
- Éléments dynamiques
- Ajax

Résultats

- Assert vs Verify
- AssertTextPresent vs AssertElementPresent vs AssertText

Localisation

- ID, name attribute, Xpath statement, Document Object Model
- Findbugs
- Objets dynamiques
- AJAX (attente d'éléments)

SELENIUM



SELENIUM



Distribué par OpenQA

licence libre Apache 2.0

8 logiciels : <http://seleniumhq.org/projects/>

<http://code.google.com/p/selenium/>

SELENIUM IDE



IDE-DemoBase - Selenium IDE 1.0.10

Fichier Édition Options ?

Base URL

Rapide Lent

Cas de test

IDE-DemoB...

Commande	Cible	Valeur
open	/	
type	q	Dominique Revuz et Etienne Duris
click	btnG	
waitForPageToLoad		5000
assertTextPresent	igm.univ-mlv.fr/~dr	
assertTextPresent	igm.univ-mlv.fr/~duris	
assertTextPresent	Ingénieurs 2000	

Commande

Cible

Valeur

Succès : 1

Échecs : 1

Log **Référence** UI-Element Combinaison

Log Référence UI-Element Combinaison Info Vider

```
[info] Executing: |open | / |
[info] Executing: |type | q | Dominique Revuz et Etienne Duris |
[info] Executing: |click | btnG |
[info] Executing: |waitForPageToLoad | | 50000 |
[info] Executing: |assertTextPresent | igm.univ-mlv.fr/~dr | |
[info] Executing: |assertTextPresent | igm.univ-mlv.fr/~duris | |
[error] false
```


SELENIUM IDE

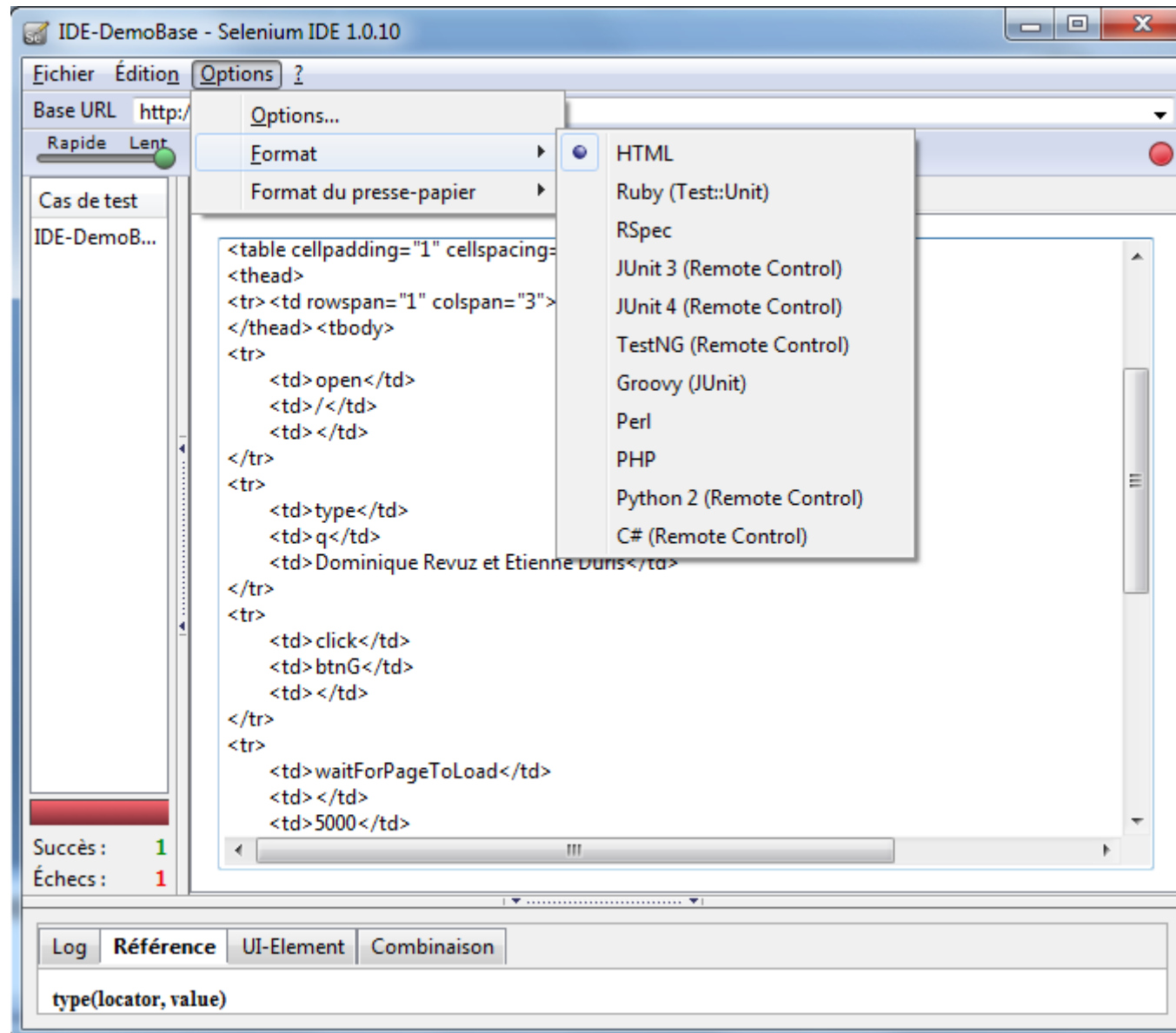


The screenshot shows the Selenium IDE 1.0.10 interface. The window title is "IDE-DemoBase - Selenium IDE 1.0.10". The menu bar includes "Fichier", "Édition", "Options", and "?". The "Base URL" is set to "http://www.google.fr/". The "Rapide" (Fast) mode is selected, and the "Lent" (Slow) mode is disabled. The "Cas de test" (Test Case) list on the left shows "IDE-DemoB...". The main area displays the source code of a test case in a table format:

Table	Source
	<pre><tr> <td> open</td> </tr> <tr> <td> type</td> <td> q</td> <td> Dominique Revuz et Etienne Duris</td> </tr> <tr> <td> click</td> <td> btnG</td> </tr> <tr> <td> waitForPageToLoad</td> <td> </td> <td> 5000</td> </tr> <tr> <td> assertTextPresent</td> <td> iam.univ-mly.fr/~dre</td> </tr></pre>

At the bottom left, the status bar shows "Succès : 1" (Success: 1) and "Échecs : 1" (Failures: 1). The bottom right corner has tabs for "Log", "Référence", "UI-Element", and "Combinaison". The "Référence" tab is active, showing the command "type(locator, value)".

SELENIUM IDE



SELENIUM IDE



The screenshot displays the Selenium IDE 1.0.10 interface. The window title is "IDE-DemoBase - Selenium IDE 1.0.10". The menu bar includes "Fichier", "Édition", "Options", and "?". The "Base URL" is set to "http://www.google.fr/". The interface is divided into several sections:

- Cas de test:** A list of test cases, with "IDE-DemoB..." visible.
- Table / Source:** The main area showing the source code for a test case. The code is as follows:

```
@Before
public void setUp() throws Exception {
    selenium = new DefaultSelenium("localhost", 4444, "*chrome", "http://www.google.fr/")
    selenium.start();
}

@Test
public void testIDE-DemoBase() throws Exception {
    selenium.open("/");
    selenium.type("q", "Dominique Revuz et Etienne Duris");
    selenium.click("btnG");
    selenium.waitForPageToLoad("");
    assertTrue(selenium.isTextPresent("igm.univ-mlv.fr/~dr"));
    assertTrue(selenium.isTextPresent("igm.univ-mlv.fr/~duris"));
    assertTrue(selenium.isTextPresent("Ingénieurs 2000"));
}

@After
public void tearDown() throws Exception {
    selenium.stop();
}
```
- Summary:** A section at the bottom left showing "Succès : 1" and "Échecs : 1".
- Log / Référence / UI-Element / Combinaison:** A section at the bottom right with tabs for "Log", "Référence", "UI-Element", and "Combinaison". The "Référence" tab is active, showing the text "type(locator, value)".

SELENIUM IDE



Sélection intelligente des champs

Commande type

Cible login_username Rechercher

Valeur

login_username	name
//input[@name='login_username']	xpath:attributes
document.forms[0].elements[0]	dom:index
UI-Element //input	xpath:position

Commande clickAndWait

Cible //input[@value='Accès Messagerie'] Rechercher

Valeur

//input[@value='Accès Messagerie']	xpath:attributes
document.forms[0].elements[4]	dom:index
//center/input	xpath:position

SELENIUM IDE



Démonstration

SELENIUM IDE



Bilan selenium IDE

Enregistrement / Lecture des tests

3 Méthodes de constructions de tests

- Enregistrement
- Ajout de vérifications (assert)
- Edition

SELENIUM IDE



Debug / Point d'arrêt

Sauvegarder/Exporter

HTML, Java, Php, Perl, Python, C#...

Plugins

SELENIUM CORE



Framework d'exécution de tests DHTML

Moteur de

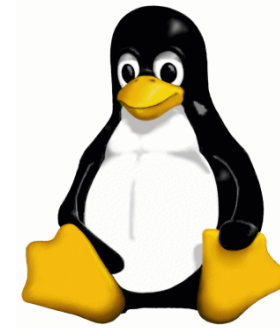
- Selenium IDE
- Selenium Remote Control

Déploiement possible sur serveur d'application

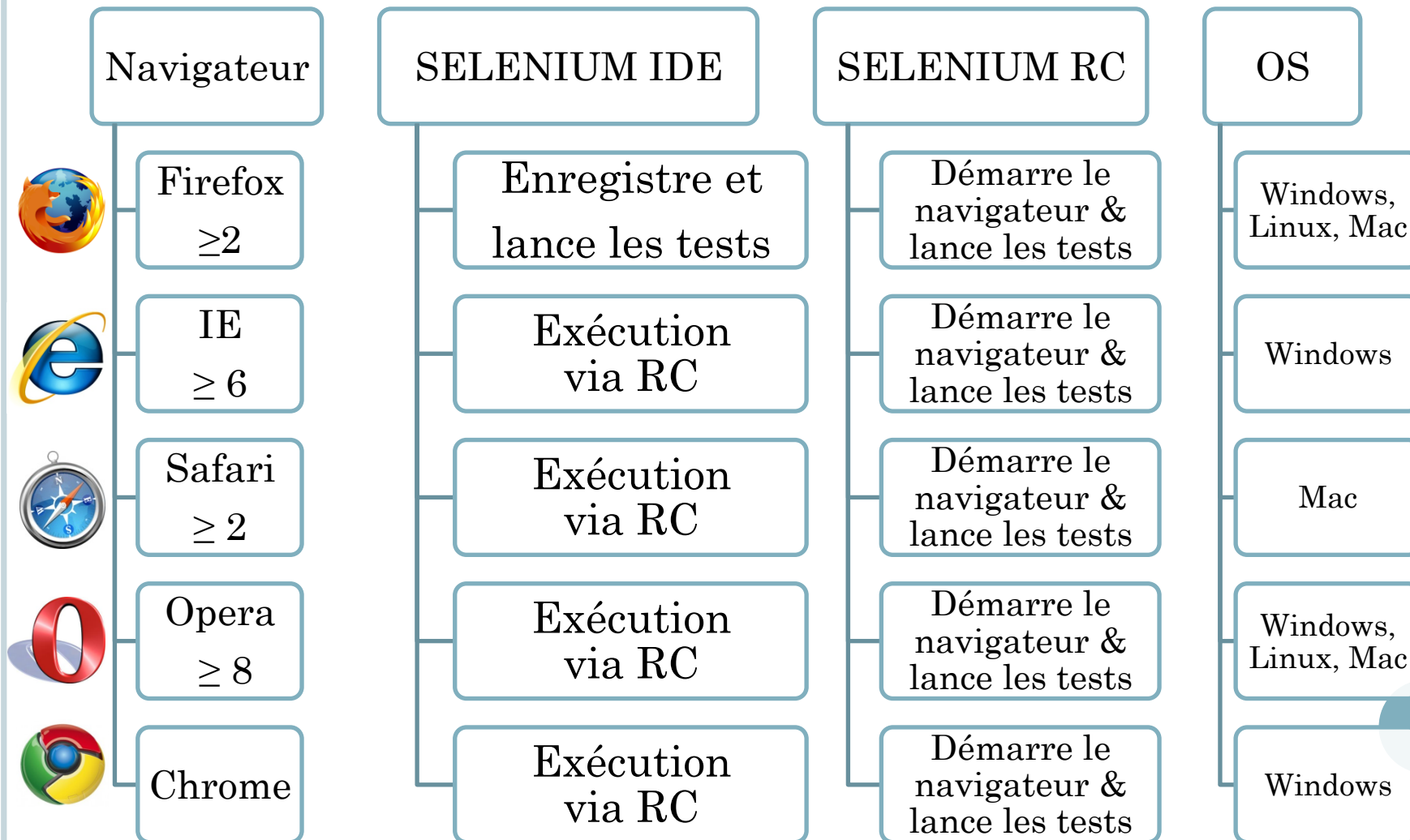
SELENIUM REMOTE CONTROL



Selenium IDE+ Core



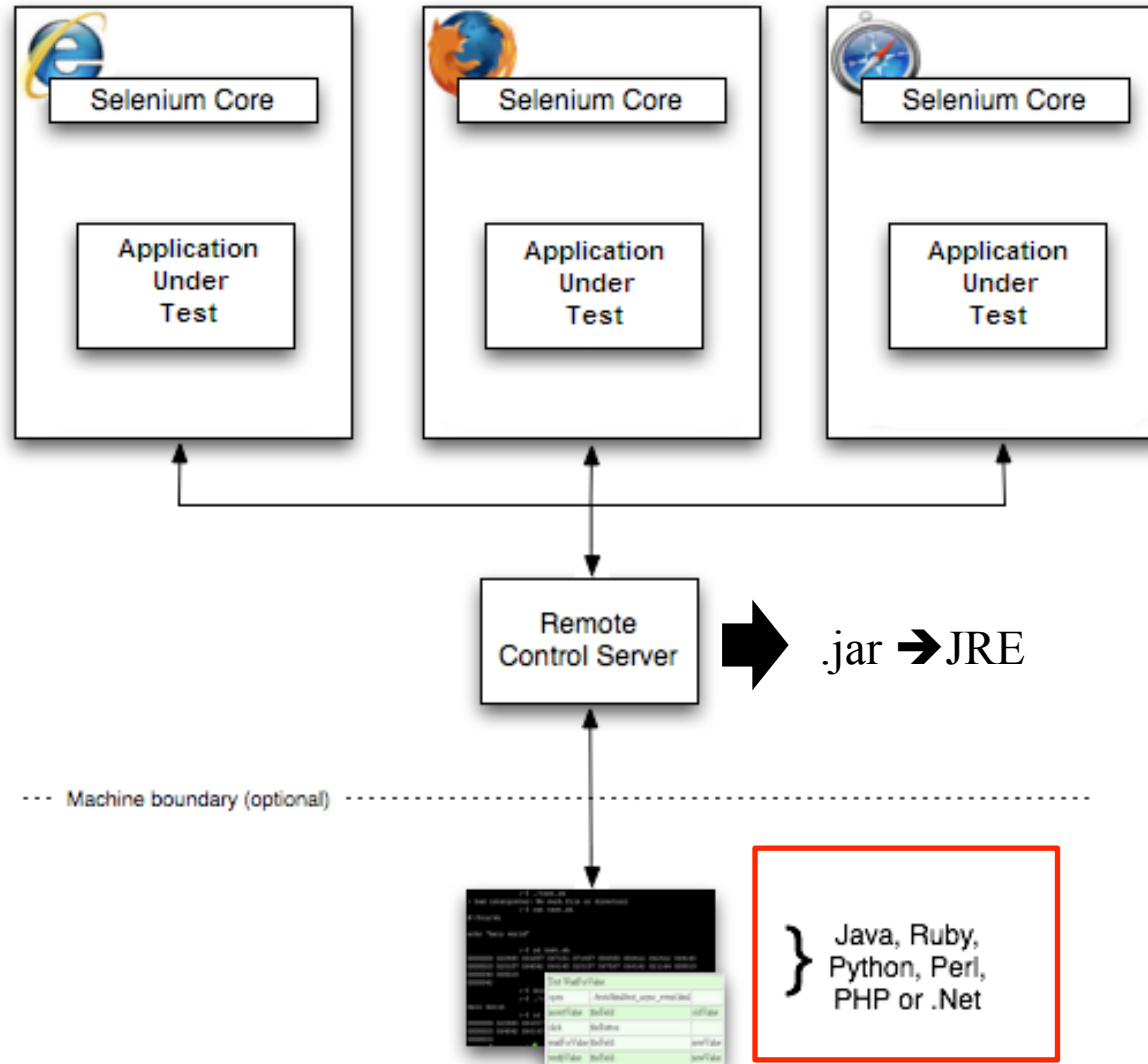
SELENIUM REMOTE CONTROL



SELENIUM REMOTE CONTROL



Windows, Linux, or Mac (as appropriate)...

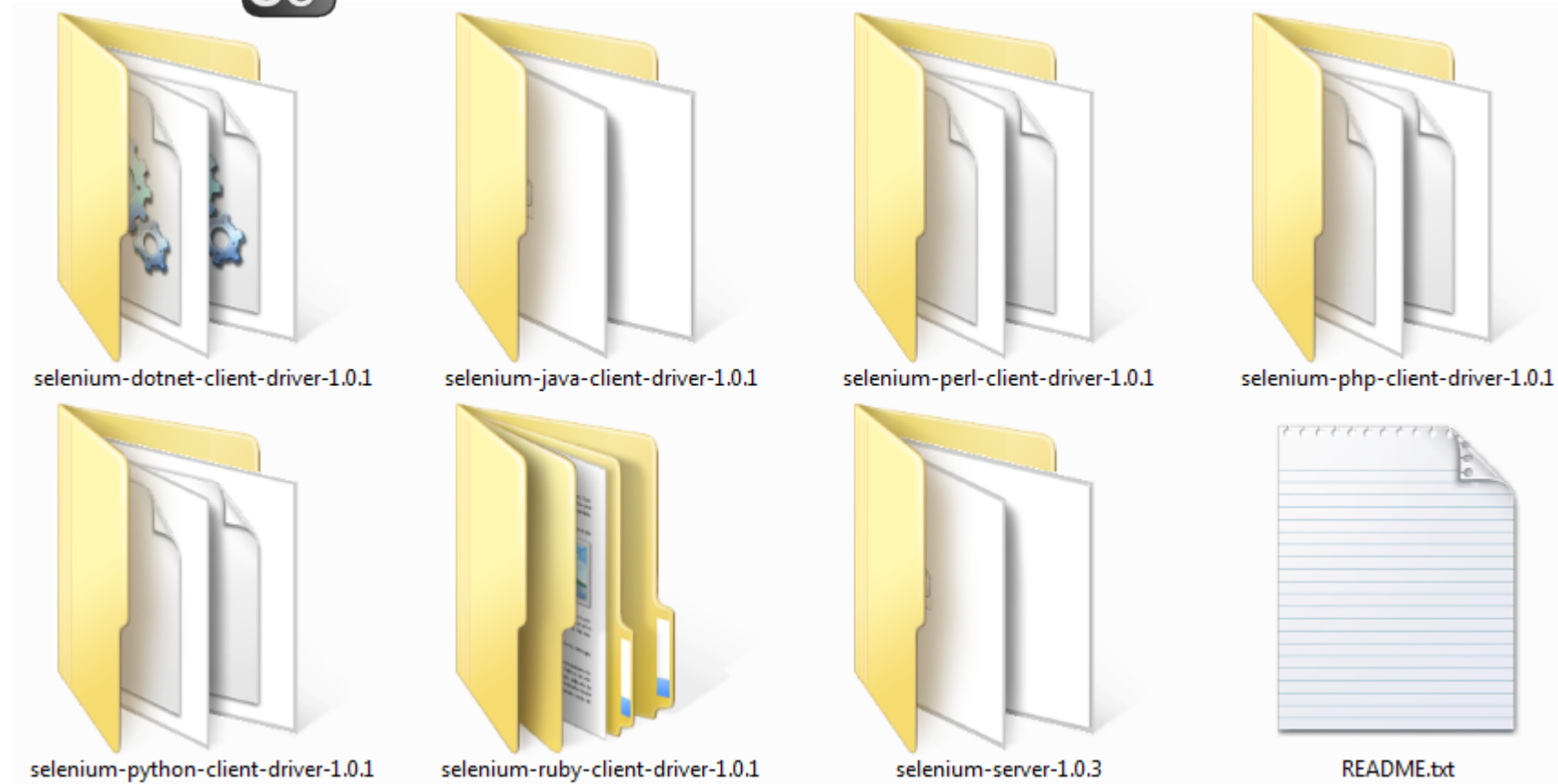


SELENIUM REMOTE CONTROL



- Condition / Itérations
- Dépendance de cas de test
- Historisation des résultats des tests (Framework)
- Gestion des erreurs
- Re-exécution des tests qui ont échoué
- Capture d'écran des tests qui ont échoués
- Tests de Base de Données

SELENIUM REMOTE CONTROL



C#

```
using System;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading;
using NUnit.Framework;
using Selenium;

namespace SeleniumTests
{
    [TestFixture]
    public class NewTest
    {
        private ISelenium selenium;
        private StringBuilder verificationErrors;

        [SetUp]
        public void SetupTest()
        {
            selenium = new DefaultSelenium("localhost", 4444, "*firefox", "http://www.google.com");
            selenium.Start();
            verificationErrors = new StringBuilder();
        }

        [TearDown]
        public void TeardownTest()
        {
            try
            {
                selenium.Stop();
            }
            catch (Exception)
            {
                // Ignore errors if unable to close the browser
            }
            Assert.AreEqual("", verificationErrors.ToString());
        }

        [Test]
        public void TheNewTest()
        {
            selenium.Open("/");
            selenium.Type("q", "selenium rc");
            selenium.Click("btnG");
            selenium.WaitForPageToLoad("30000");
            Assert.AreEqual("selenium rc - Google Search", selenium.GetTitle());
        }
    }
}
```

2. Selenium

```
/** Add JUnit framework to your classpath if not already there
 * for this example to work
 */
package com.example.tests;

import com.thoughtworks.selenium.*;
import java.util.regex.Pattern;

public class NewTest extends SeleneseTestCase {
    public void setUp() throws Exception {
        setUp("http://www.google.com/", "*firefox");
    }
    public void testNew() throws Exception {
        selenium.open("/");
        selenium.type("q", "selenium rc");
        selenium.click("btnG");
        selenium.waitForPageToLoad("30000");
        assertTrue(selenium.isTextPresent("Results * for selenium rc"));
    }
}
```

Perl

```
/** Add JUnit framework to your classpath if not already there
 * for this example to work
 */
package com.example.tests;

import com.thoughtworks.selenium.*;
import java.util.regex.Pattern;

public class NewTest extends SeleneseTestCase {
    public void setUp() throws Exception {
        setUp("http://www.google.com/", "*firefox");
    }
    public void testNew() throws Exception {
        selenium.open("/");
        selenium.type("q", "selenium rc");
        selenium.click("btnG");
        selenium.waitForPageToLoad("30000");
        assertTrue(selenium.isTextPresent("Results * for selenium rc"));
    }
}
```



```
<?php

require_once 'PHPUnit/Extensions/SeleniumTestCase.php';

class Example extends PHPUnit_Extensions_SeleniumTestCase
{
    function setUp()
    {
        $this->setBrowser("*firefox");
        $this->setBrowserUrl("http://www.google.com/");
    }

    function testMyTestCase()
    {
        $this->open("/");
        $this->type("q", "selenium rc");
        $this->click("btnG");
        $this->waitForPageToLoad("30000");
        $this->assertTrue($this->isTextPresent("Results * for selenium rc"));
    }
}
?>
```

```
from selenium import selenium
import unittest, time, re

class NewTest(unittest.TestCase):
    def setUp(self):
        self.verifcationErrors = []
        self.selenium = selenium("localhost", 4444, "*firefox",
                                "http://www.google.com/")
        self.selenium.start()

    def test_new(self):
        sel = self.selenium
        sel.open("/")
        sel.type("q", "selenium rc")
        sel.click("btnG")
        sel.wait_for_page_to_load("30000")
        self.failUnless(sel.is_text_present("Results * for selenium rc"))

    def tearDown(self):
        self.selenium.stop()
        self.assertEqual([], self.verifcationErrors)
```

```
require "selenium"
require "test/unit"

class NewTest < Test::Unit::TestCase
  def setup
    @verification_errors = []
    if $selenium
      @selenium = $selenium
    else
      @selenium = Selenium::SeleniumDriver.new("localhost", 4444, "selenium")
      @selenium.start
    end
    @selenium.set_context("test_new")
  end

  def teardown
    @selenium.stop unless $selenium
    assert_equal [], @verification_errors
  end

  def test_new
    @selenium.open "/"
    @selenium.type "q", "selenium rc"
    @selenium.click "btnG"
    @selenium.wait_for_page_to_load "30000"
    assert @selenium.is_text_present("Results * for selenium rc")
  end
end
```

SELENIUM REMOTE CONTROL



Options du serveur :

- Configurer le proxy
- Mode Mono/Multi fenêtre(s)
- Choix du profil Firefox
- Exécuter directement du HTML
- Récupérer les logs directement
- Chemin absolu vers le navigateur (ex : si plusieurs versions)
- Mode interactif

Plus d'options en tapant : `java -jar selenium-server.jar -h`

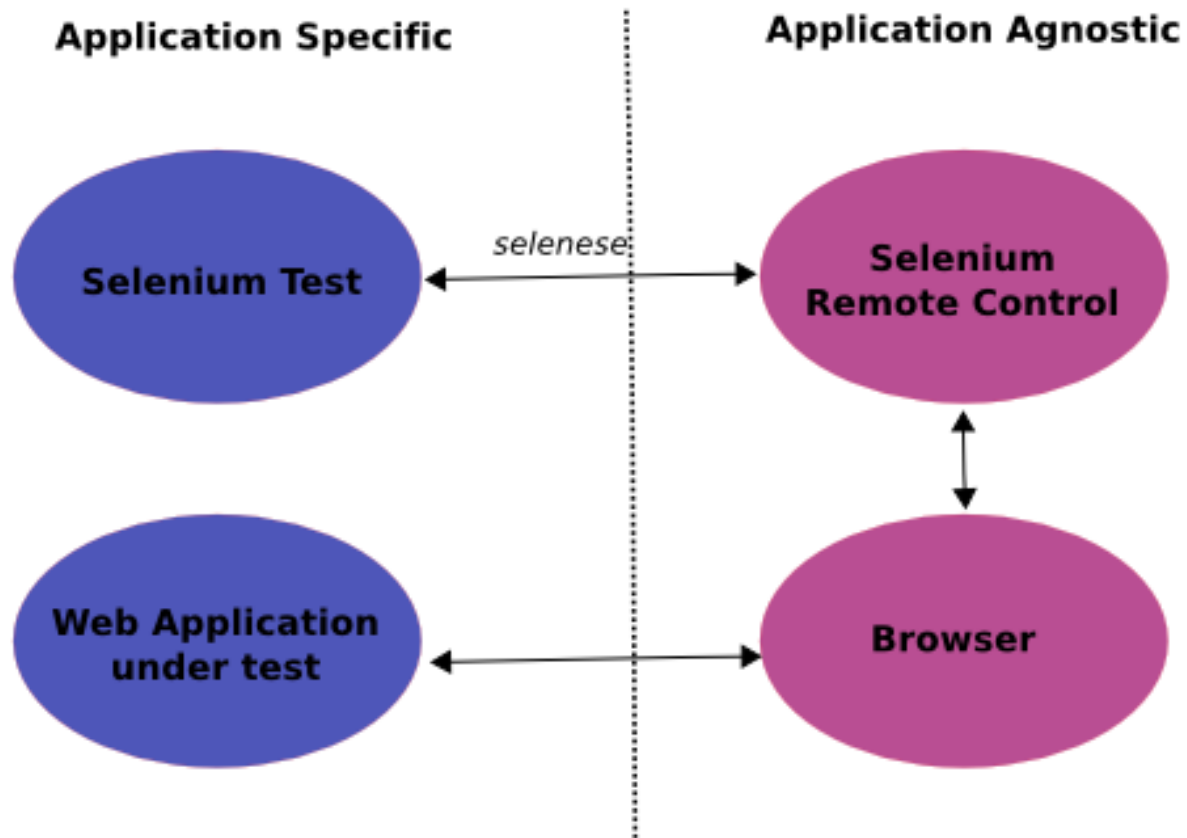
SELENIUM GRID



Limitations de RC :

- Tests concurrents limités (<6 conseillé / RC)
- Si trop rapide : Goulot d'étranglement
Selenium RC ⇔ navigateur
- Tests différentes versions des navigateurs & OS.
→ Architecture compliqué et non maintenable

SELENIUM GRID

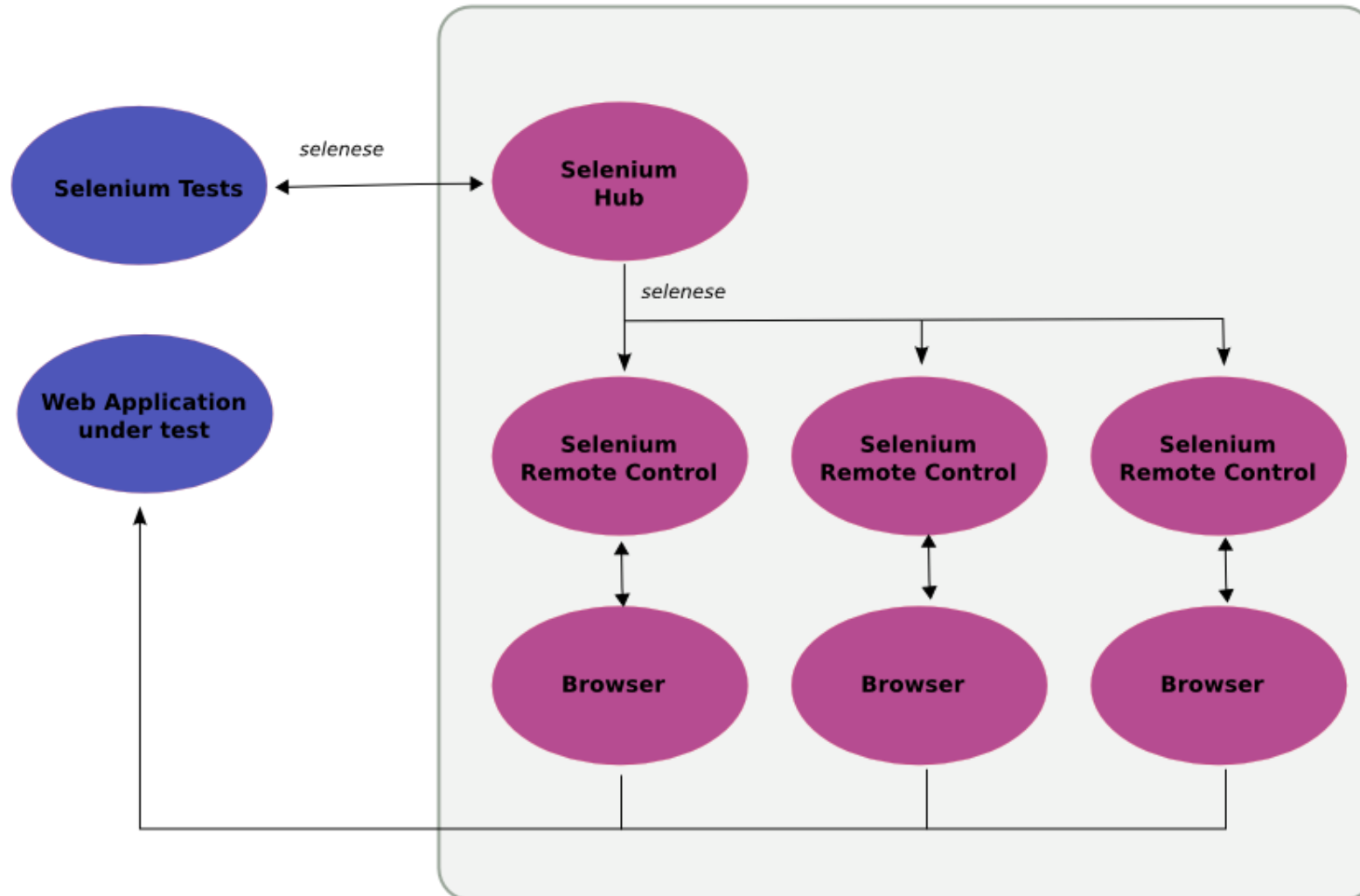


SELENIUM GRID

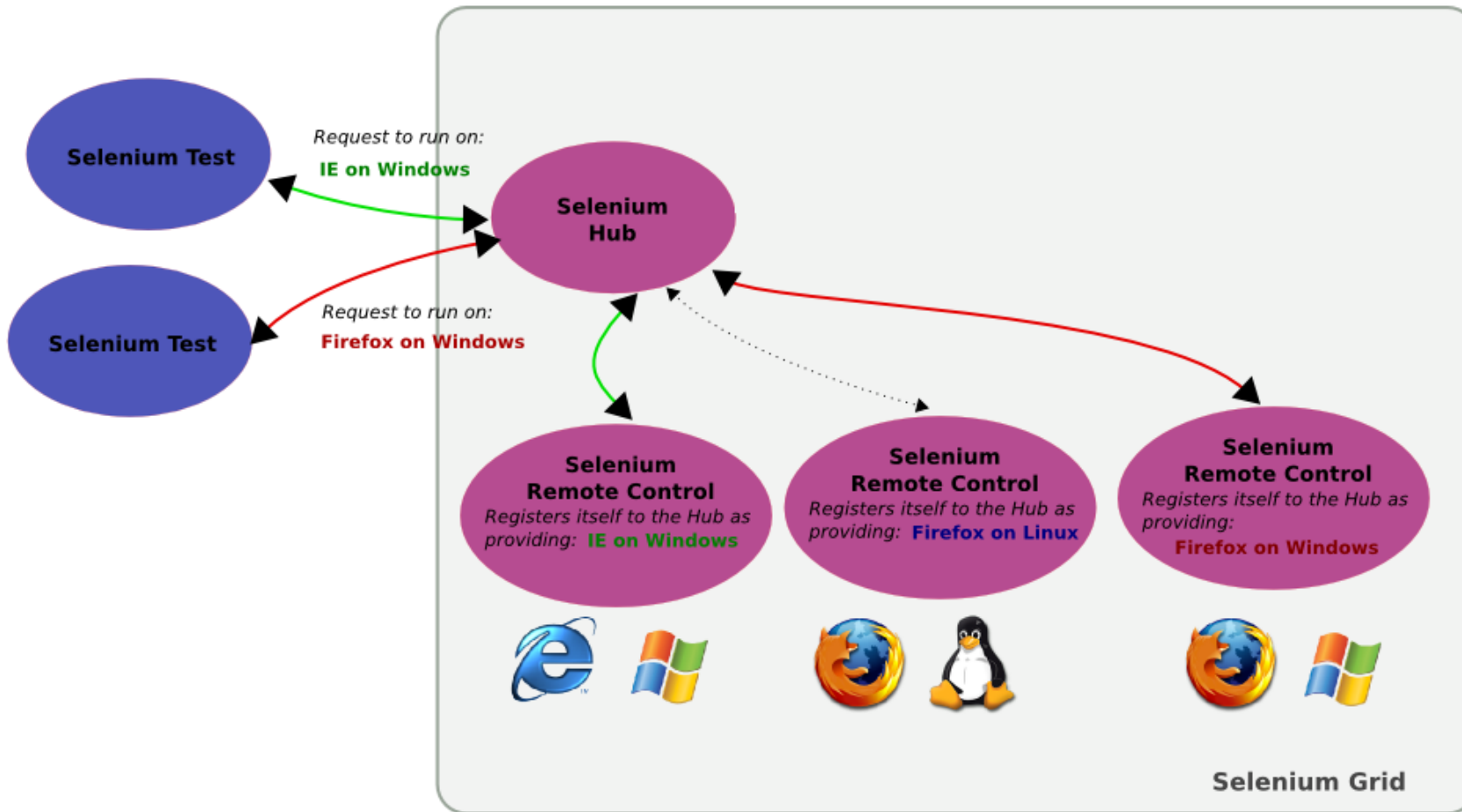


Application Specific

Selenium Grid - Application Agnostic



SELENIUM GRID



SELENIUM GRID



Avantages :

- Améliorations des performances
- Limitation de l'utilisation de chaque RC
- Choix du RC transparent (hub)
- Tests multi navigateurs & multi OS faciles

Exemple en java

```
new DefaultSelenium("localhost", 4444, ***firefox***, 'http://www.google.fr');
```

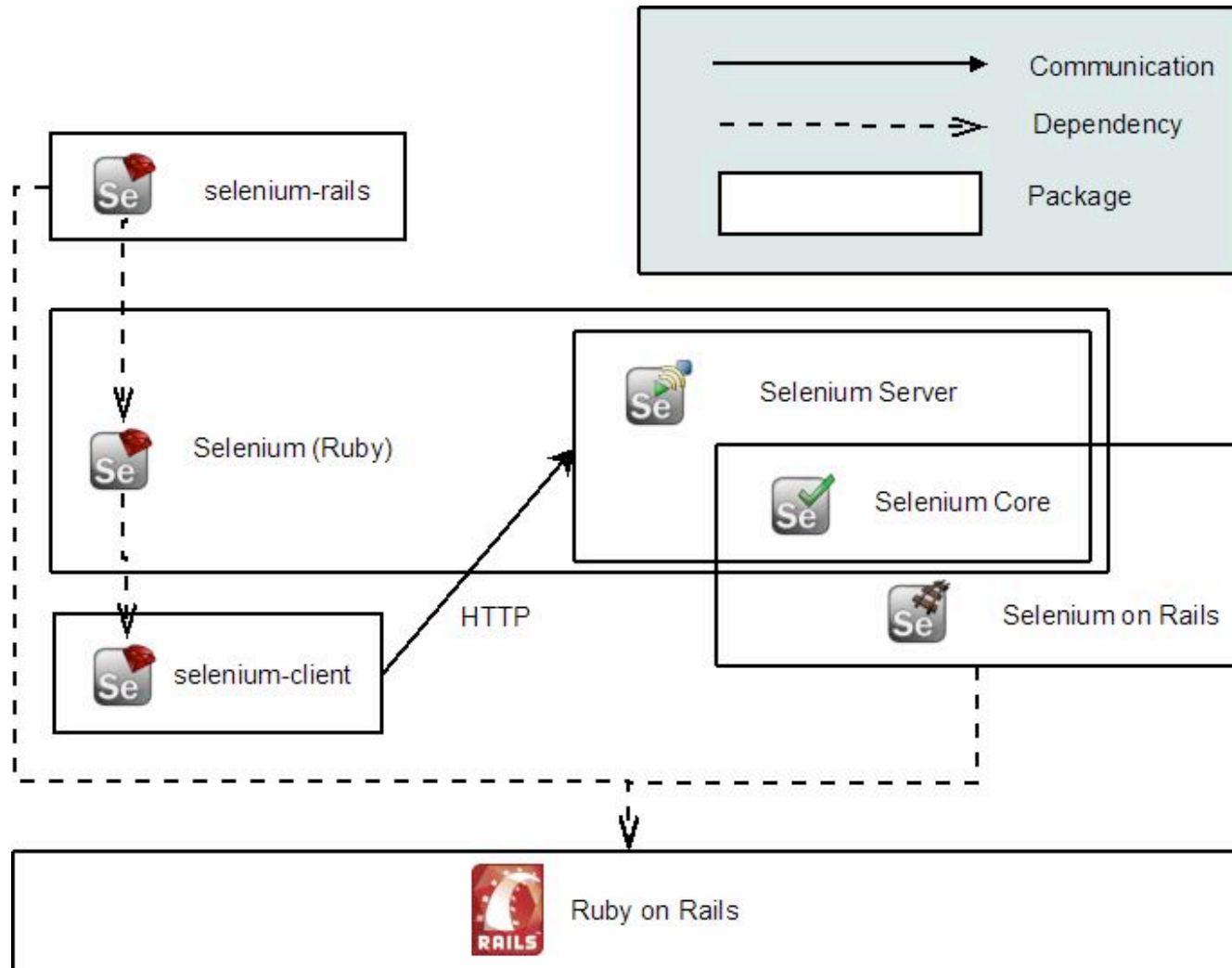
devient

```
new DefaultSelenium("hub", 4444, **Chrome on Ubuntu**, 'http://www.google.fr');
```

```
new DefaultSelenium("hub", 4444, **IE 7 on Chinese locale**, 'http://www.google.fr');
```

```
new DefaultSelenium("hub", 4444, **Safari on Quadri-processor**, 'http://www.google.fr');
```

SELENIUM ON RUBY & ON RAILS

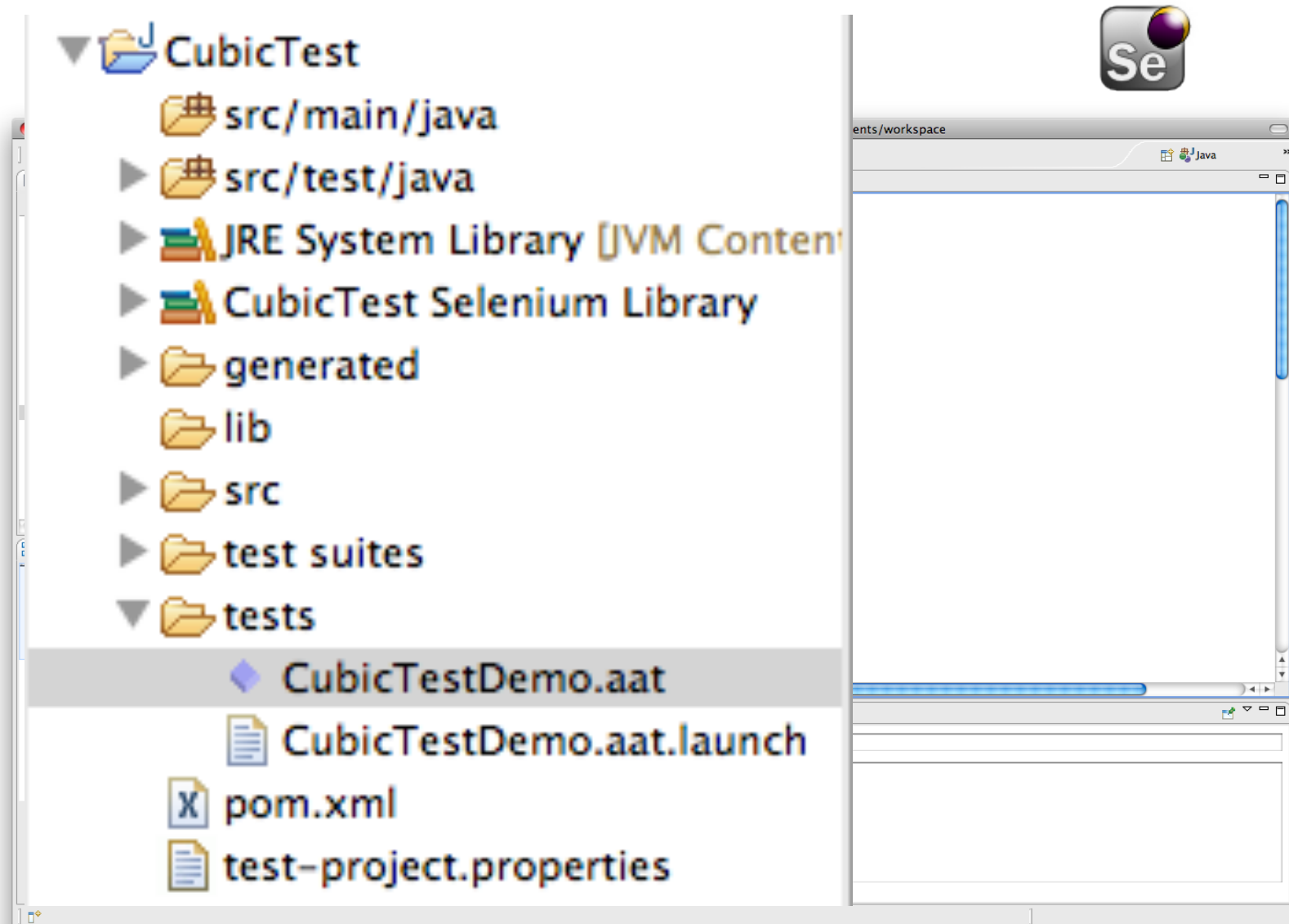


CUBIC TEST (POUR ECLIPSE)

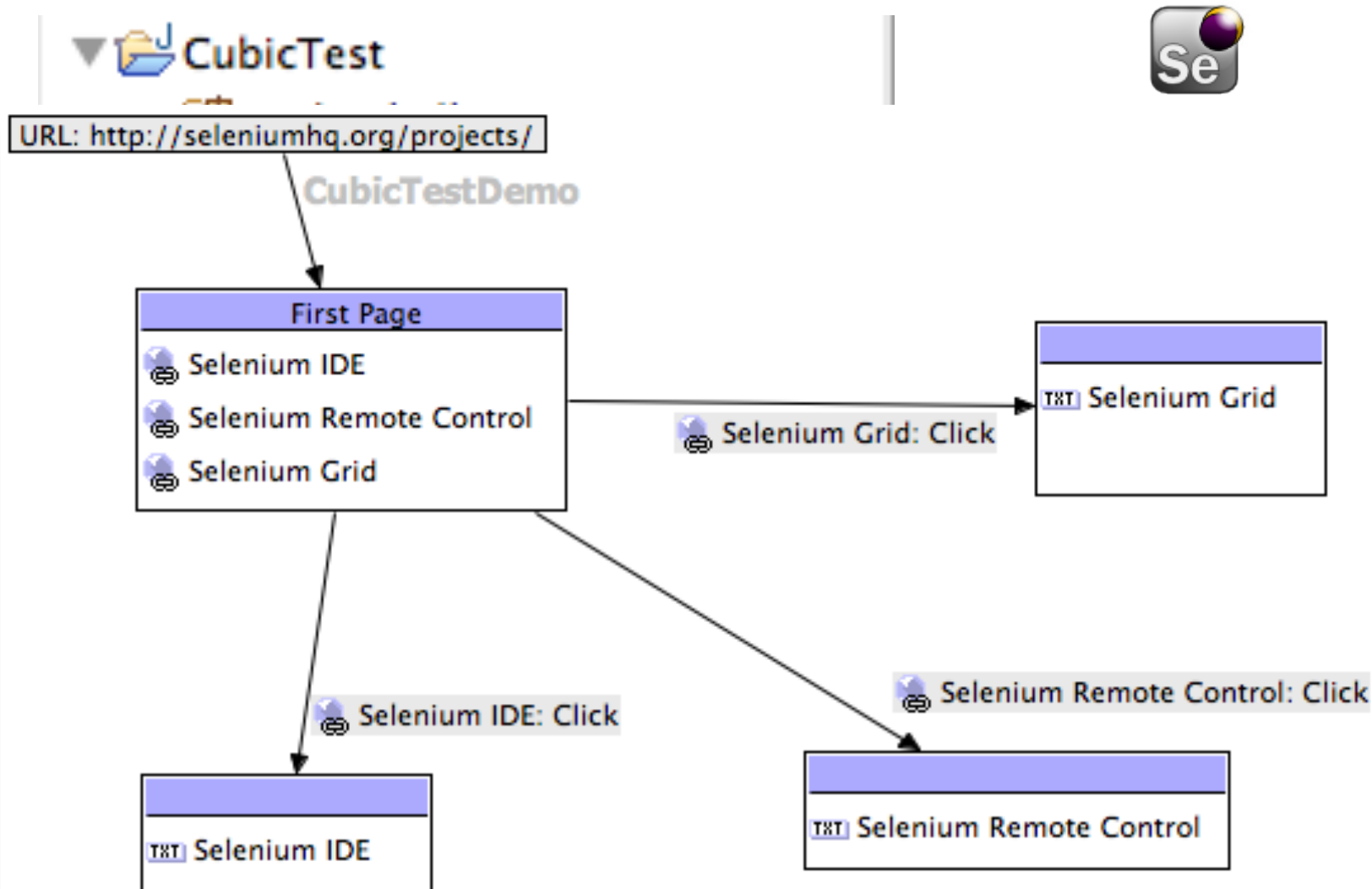


The screenshot shows the Eclipse IDE with the Selenium IDE plugin. The main editor displays a test suite diagram for 'CubicTestDemo' with the URL 'http://seleniumhq.org/projects/'. The diagram shows a 'First Page' containing three Selenium IDE elements: 'Selenium IDE', 'Selenium Remote Control', and 'Selenium Grid'. The 'Selenium IDE' element is connected to 'Selenium Remote Control' and 'Selenium Grid'. The 'Selenium Remote Control' element is connected to 'Selenium IDE' and 'Selenium Remote Control'. The 'Selenium Grid' element is connected to 'Selenium Remote Control'. The Package Explorer on the left shows the project structure, and the Properties window at the bottom shows the test name 'CubicTestDemo'.

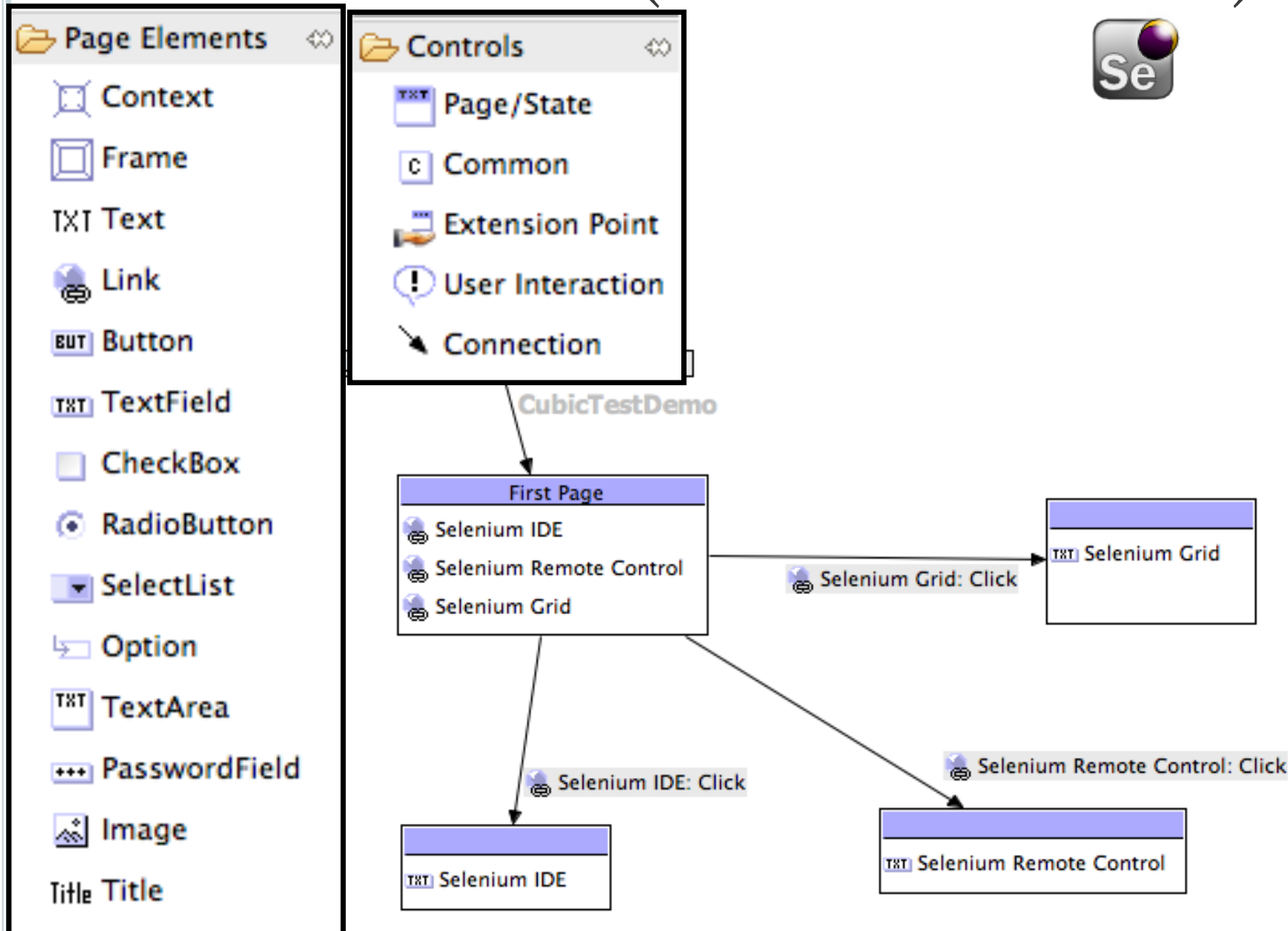
CUBIC TEST (POUR ECLIPSE)



CUBIC TEST (POUR ECLIPSE)



CUBIC TEST (POUR ECLIPSE)



CUBIC TEST (POUR ECLIPSE)



The screenshot shows the Eclipse IDE interface. A context menu is open over a Selenium Grid element. The menu items are:

- Cut
- Copy
- Paste
- Change Start Point for Test
- Reset Test-Run
- Auto-Layout
- Export to Watir test script
- Export to Selenium Core test script
- Run Test with Watir Runner
- Run As** (highlighted)
- Debug As
- Profile As
- Validate
- Team
- Compare With
- Replace With
- WikiText
- Remove from Context

The 'Run As' submenu is open, showing the following options:

- 1 Record Test with Selenium
- 2 Run Test with Selenium
- Run Configurations...

CUBIC TEST (POUR ECLIPSE)



Fonctionnalités de IDE + Core + RC (php et java)

Couche Graphique

Watir ?? (Partie 3)

BROMINE



A snippet of the periodic table showing elements Selenium (Se), Bromine (Br), and Krypton (Kr). The table is tilted and includes atomic numbers, element symbols, names, atomic weights, and electron configurations.

Atomic Number	Element Symbol	Element Name	Atomic Weight	Electron Configuration
34	Se	Selenium	78.96	$3d^{10} 4s^2 4p^4$
35	Br	Bromine	79.904	$[Ar] 3d^{10} 4s^2 4p^5$
36	Kr	Krypton	83.798	$[Ar] 3d^{10} 4s^2 4p^6$

BROMINE

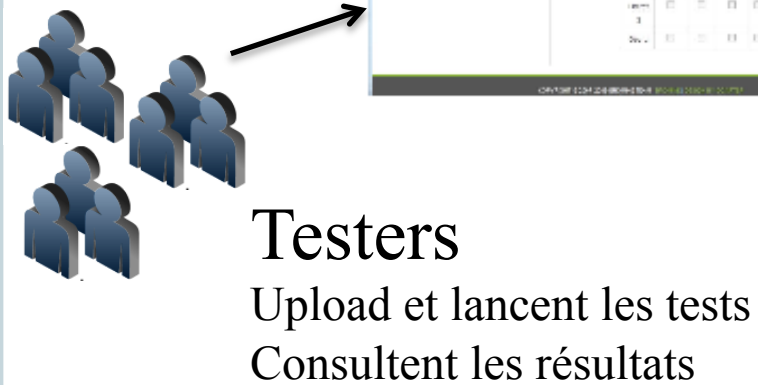
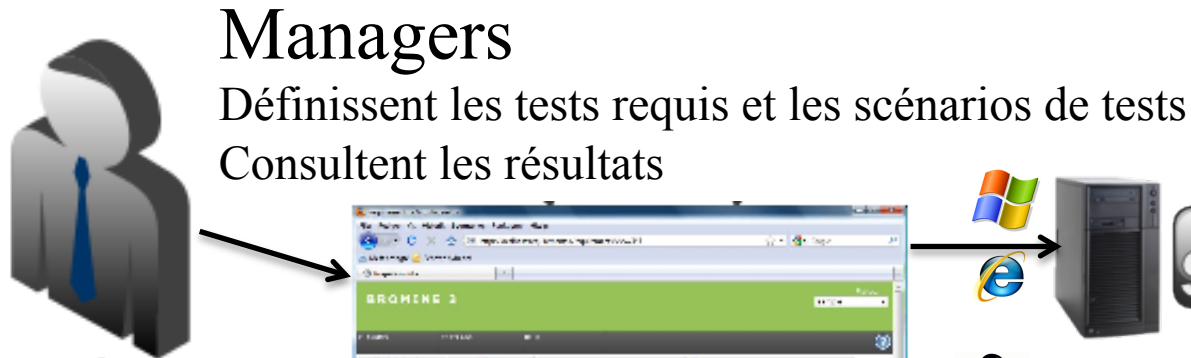
Reporting : utiliser des Frameworks

- Java
 - JUnit Report
 - TestNG Report
 - ReportNG
 - TestNG-xslt
- Python
 - HTMLTestRunner
- Ruby
 - RSpec Report

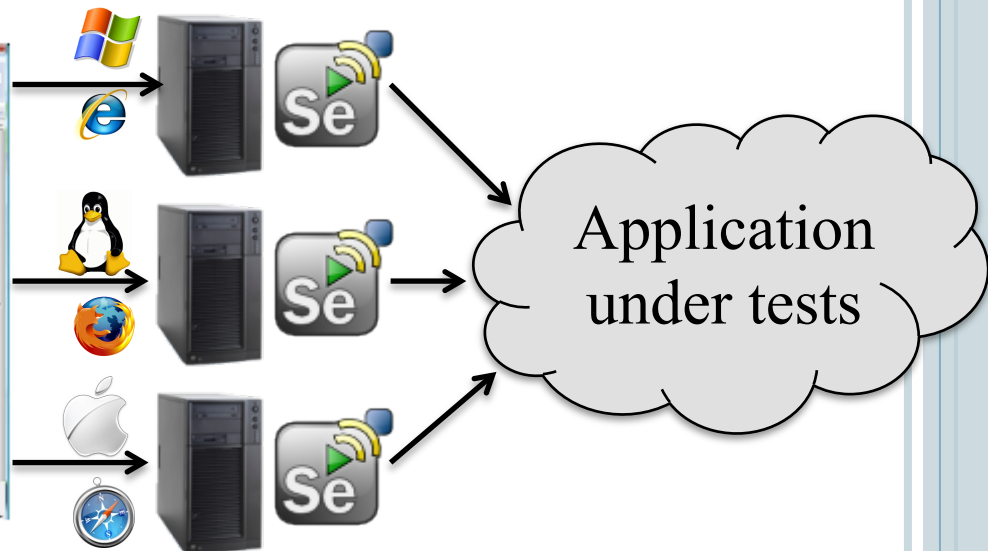


Ou : utiliser Bromine !

BROMINE



Page	description	Statut	Pass	Fail	Not Run	Blocked
Page 1	Page 1	Pass	1	0	0	0
Page 2	Page 2	Pass	1	0	0	0
Page 3	Page 3	Pass	1	0	0	0
Page 4	Page 4	Pass	1	0	0	0
Page 5	Page 5	Pass	1	0	0	0
Page 6	Page 6	Pass	1	0	0	0
Page 7	Page 7	Pass	1	0	0	0
Page 8	Page 8	Pass	1	0	0	0
Page 9	Page 9	Pass	1	0	0	0
Page 10	Page 10	Pass	1	0	0	0



BROMINE



- Tests en Java et Php
- Enregistrement possible avec IDE
- Upload de tests
- Tests multi navigateurs & OS en un click
- RCs sous forme de Nodes (sans GRID)
- Logs des résultats
- Groupes ↔ Projets
- Plugins

SELENIUM : BILAN



SELENIUM : BILAN



Fonctions
de bases

Utilisateurs
de Ruby

Tests multi
plateformes/OS

Approche
projet

Couche
graphique

WATIR

Open source  open source

Communauté active et grandissante

Ruby 

Multi plateforme / OS



"Powerfull"

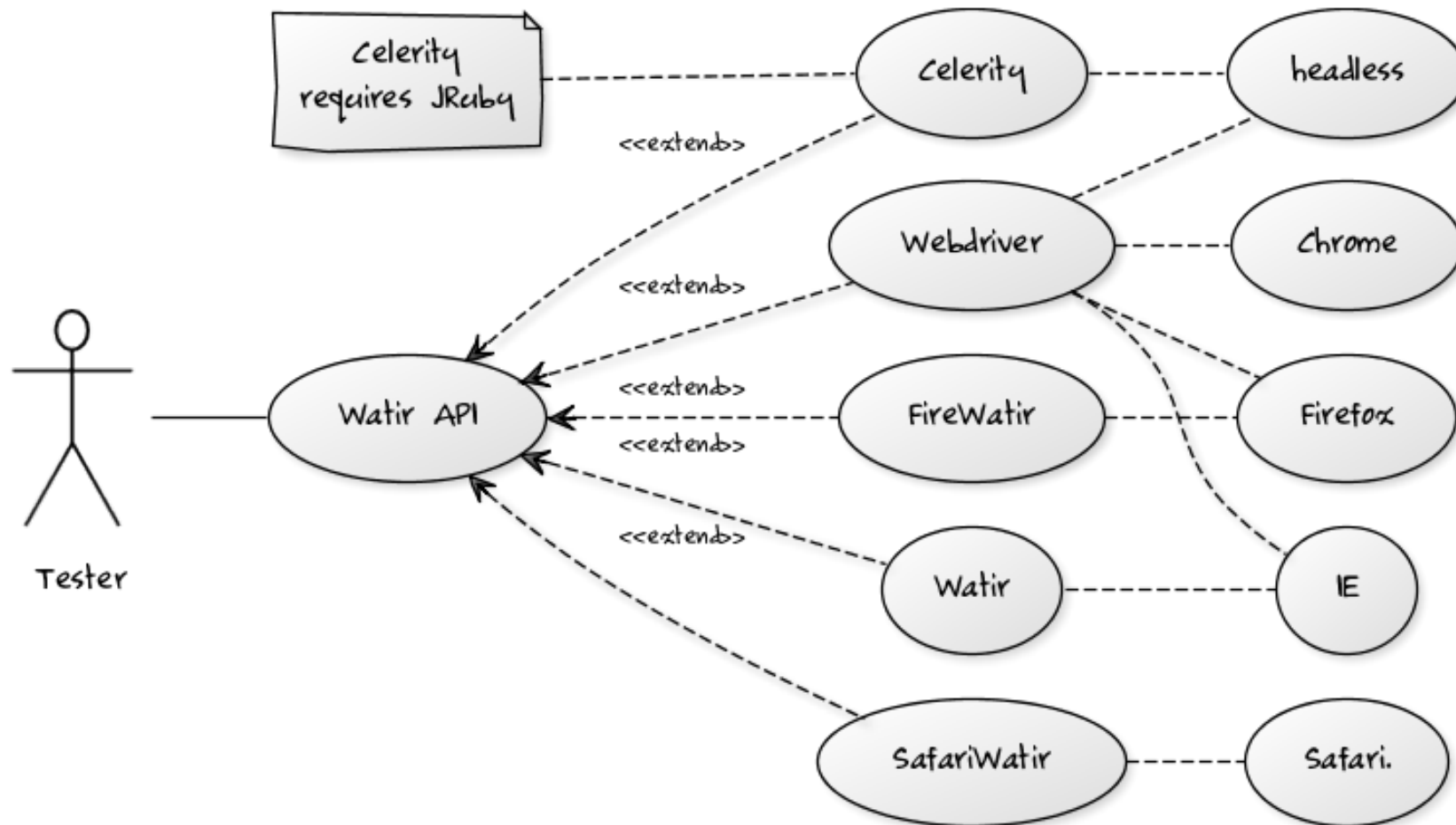
"easy to use"

"beautifully lightweight"

WATIR

Automated testing that doesn't hurt

WATIR



#sudo gem install xxxxxxxx

CONCLUSION 1/2

Nouvelle approche des tests d'interface web

Mon choix :

Selenium

- Adapté à différents niveaux de besoins
- Multi langages

Mon avis :

Watir

- Adapté à un seul un seul niveau de besoin
- « La terre ne s'est pas faite en un jour »

CONCLUSION 2/2

Selenium : Compatible avec Spring Roo

Des betas coté Selenium...

- Tests sous Android
- Selenium Standalone
(RC + WebDriver en un jar)
- Selenium 2
(WebDriver)

<http://code.google.com/p/selenium/downloads/list>

QUESTIONS ?



Le Chat de Philippe Geluck