

INTERMEDIO

Unidad 02

Selenium

→ CURSOS DE
DESARROLLO

Objetivos

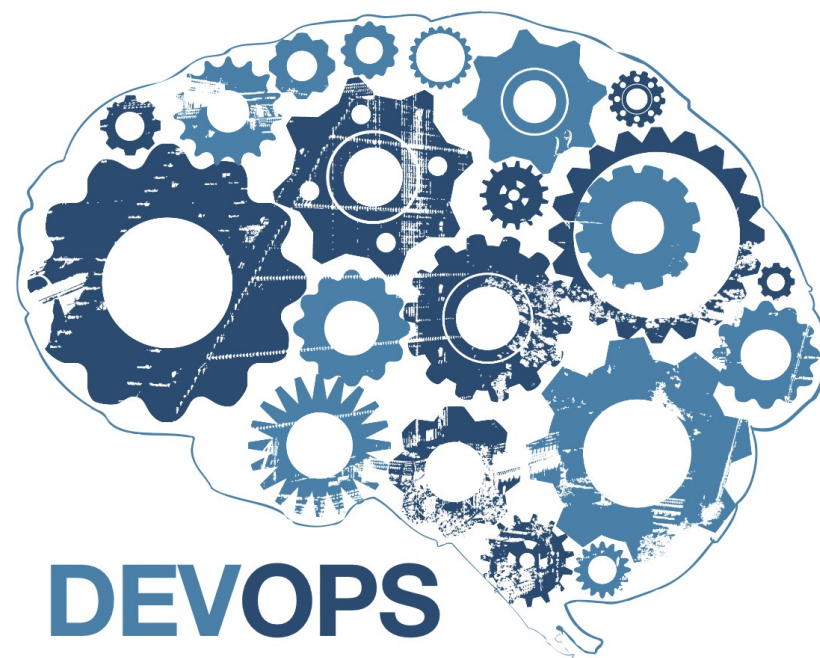
¿Qué voy a aprender?



- Saber para qué sirven las pruebas de aceptación
- Saber buscar elementos en el DOM
- Saber cómo lanzar las pruebas

Contenidos

¿Cómo voy a aprenderlo?



- 1.Introducción
- 2.Entorno de trabajo
- 3.Características
- 4.Sintaxis del lenguaje de script
- 5.Construcción y ejecución de pruebas
- 6.Pruebas de Selenium desde Java
- 7.API Selenium
- 8.Selenium Grid

Introducción

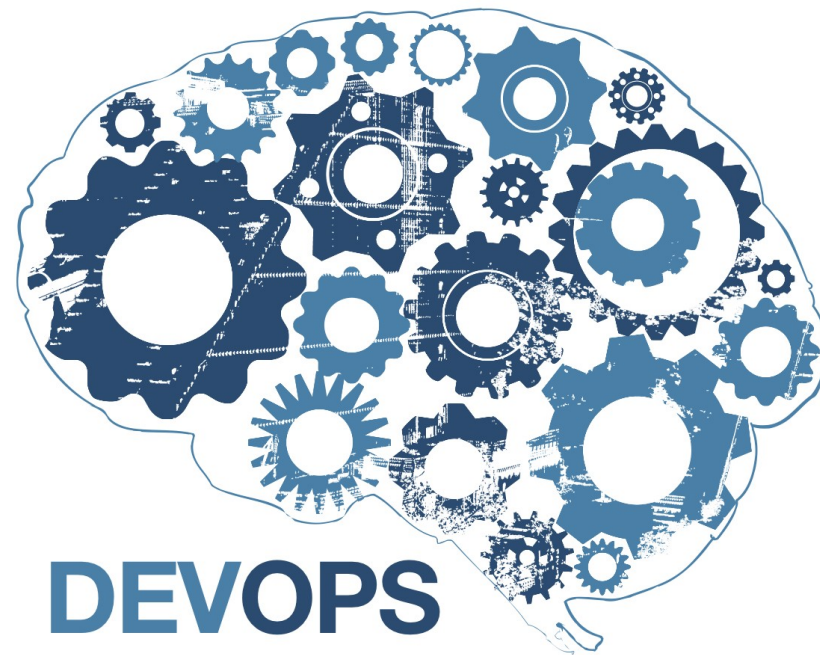
¿Por dónde empezamos?



Una de las problemáticas principales a la hora de probar una web es la automatización de las pruebas

Introducción

¿Por dónde empezamos?



Es muy complicado hacerlas y afinarlas, por lo que necesitamos apoyarnos en algún tipo de Framework que nos permita programarlas

Introducción

¿Por dónde empezamos?

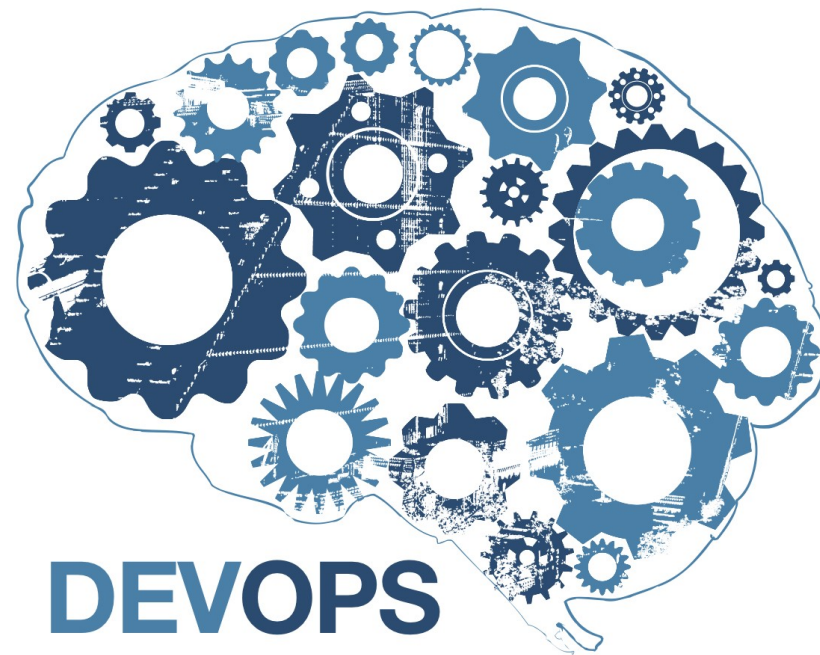


Selenium es uno de los frameworks más avanzados a la hora de definir y programar pruebas

<http://www.seleniumhq.org/>

Entorno de Trabajo

¿Cómo las desarrollo?



- Selenium IDE: Plugin de Firefox y Chrome
- Selenium WebDriver: Simulación de un Navegador Web
- Selenium Server/Hub/Grid: Servidor de ejecución de pruebas en paralelo

Entorno de Trabajo

¿Cómo las desarrollo?



Selenium IDE

Permite la grabación de procesos en la web
desde Firefox y Chrome

<http://www.seleniumhq.org/projects/ide/>

Entorno de Trabajo

¿Cómo las desarrollo?



Selenium WebDriver

Permite ejecución de pruebas mediante un lenguaje de programación

http://www.seleniumhq.org/docs/03_webdriver.jsp

Entorno de Trabajo

¿Cómo las desarrollo?



Selenium Grid

Permite ejecución de pruebas remotas mediante un servidor de pruebas

http://www.seleniumhq.org/docs/07_selenium_grid.jsp

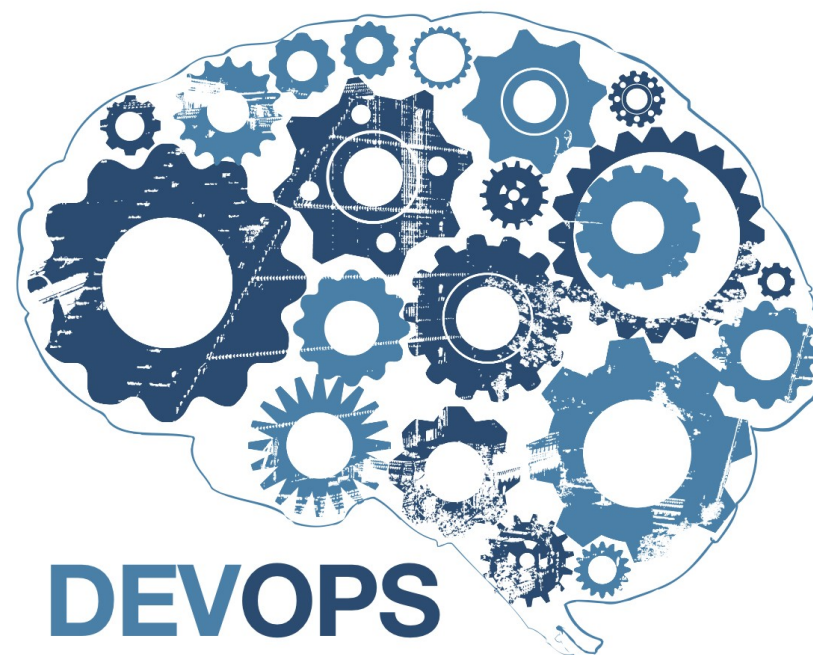
Características

¿Qué funcionalidades tiene?

Selenium WebDriver

Permite la ejecución de pruebas en base a un Navegador Web

- Google Chrome
- Internet Explorer 6, 7, 8, 9, 10 - 32 and 64-bit where applicable
- Firefox: latest ESR, previous ESR, current release, one previous release
- Safari
- Opera
- HtmlUnit
- phantomjs
- Android (with Selendroid or appium)
- iOS (with ios-driver or appium)



Características

¿Qué funcionalidades tiene?



Selenium WebDriver

Permite la ejecución de pruebas desde:

- Java
- C#
- Python
- Ruby
- PHP
- Perl
- Javascript

Sintaxis

¿Qué estructura tiene?



Comandos típicos:

open

click

verifyTitle

verifyTextPresent

verifyElementPresent

waitForPageToLoad

waitForElementPresent

Sintaxis

¿Qué estructura tiene?



Localizadores:

id

nombre

XPath

link

DOMElement

CSS

Construcción y Ejecución

¿Cómo los guardo y lanzo?



Las pruebas habitualmente se graban con Selenium
IDE

Se pueden guardar en ficheros .html
Después puedes exportarse a JUnit
o cualquier otro motor de pruebas

Construcción y Ejecución

¿Cómo los guardo y lanzo?



Instalación de Selenium IDE

<https://www.seleniumhq.org/selenium-ide/>

<https://github.com/SeleniumHQ/selenium/wiki/SeIDE-Release-Notes>

Construcción y Ejecución

¿Cómo los guardo y lanzo?



Tipos de comandos:

Actions: acciones que realiza el usuario

Accessors: Localizadores de elementos

Assertions: comprobaciones de contenido

Construcción y Ejecución

¿Cómo los guardo y lanzo?



Uso Básico de Selenium IDE
Tutorial básico

<http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/381>

Construcción y Ejecución

¿Cómo los guardo y lanzo?



Demo de Selenium IDE

Construcción y Ejecución

¿Cómo los guardo y lanzo?



Una vez exportadas las pruebas pueden retocarse desde un IDE

Pruebas desde Java

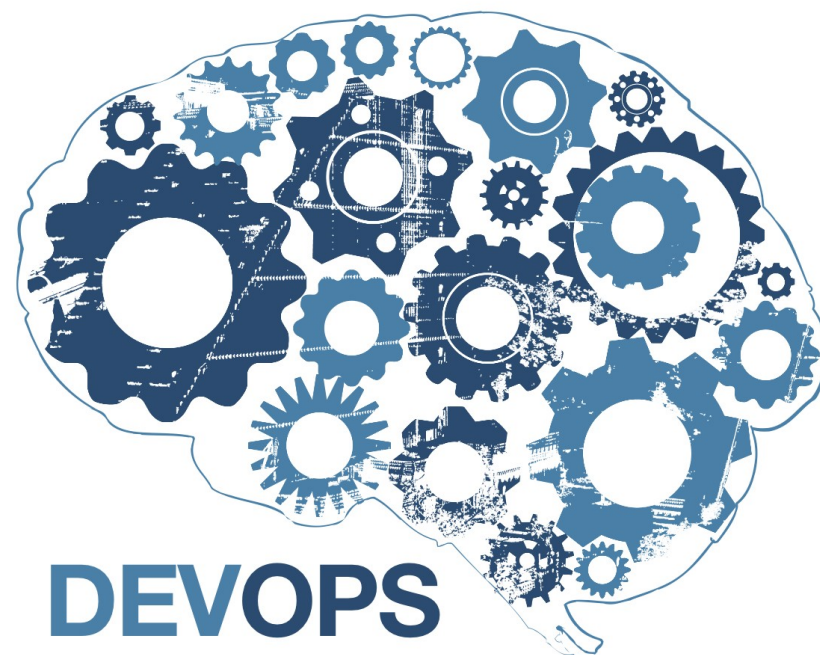
¿Cómo los compilo y edito?



```
<dependency>  
<groupId>org.seleniumhq.selenium</groupId>  
<artifactId>selenium-java</artifactId>  
<version>2.46.0</version>  
<scope>test</scope>  
</dependency>
```

Pruebas desde Java

¿Cómo los compilo y edito?



```
WebDriver driver = new FirefoxDriver();
driver.get("http://www.google.com");
WebElement element = driver.findElement(By.name("q"));
element.sendKeys("Cheese!");
element.submit();
System.out.println("Page title is: " + driver.getTitle());
(new WebDriverWait(driver, 10)).until(new
ExpectedCondition<Boolean>() {
    public Boolean apply(WebDriver d) {
        return d.getTitle().toLowerCase().startsWith("cheese!");
    }
});
System.out.println("Page title is: " + driver.getTitle());
driver.quit();
```


Selenium API

¿Cómo los compilo y edito?



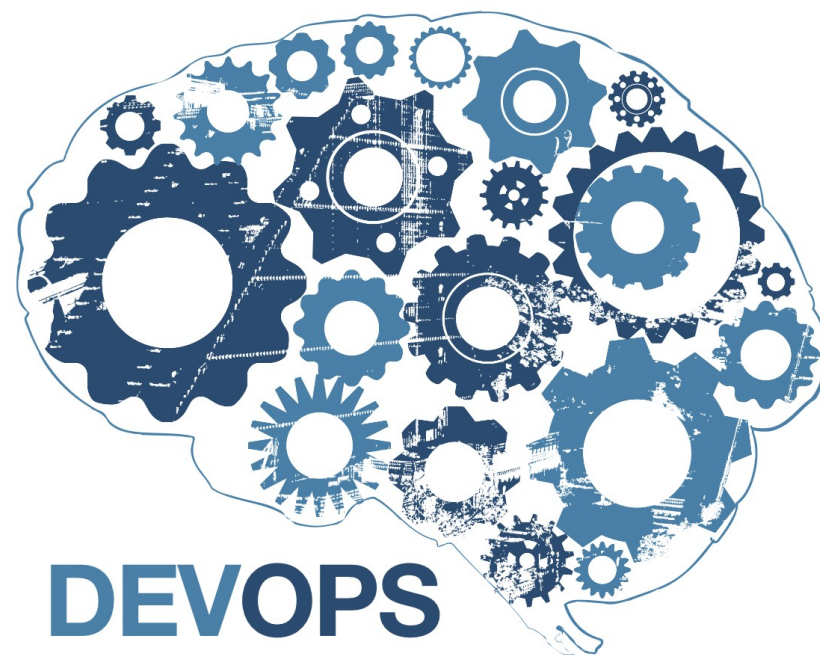
Localizadores:

Clase By

- id
- className
- tagName
- linkText
- partialLinkText
- cssSelector
- xpath

Selenium API

¿Cómo los compilo y edito?



Ejemplo de localizador:
WebElement element =
driver.findElement(By.id("coolestWidgetEvah"));

Selenium API

¿Cómo los compilo y edito?



Formularios:

- select
- deselectAll
- selectByVisibleText

Campos

- value

Botones

- click
- submit

Selenium Grid

¿Cómo lanzo en paralelo?

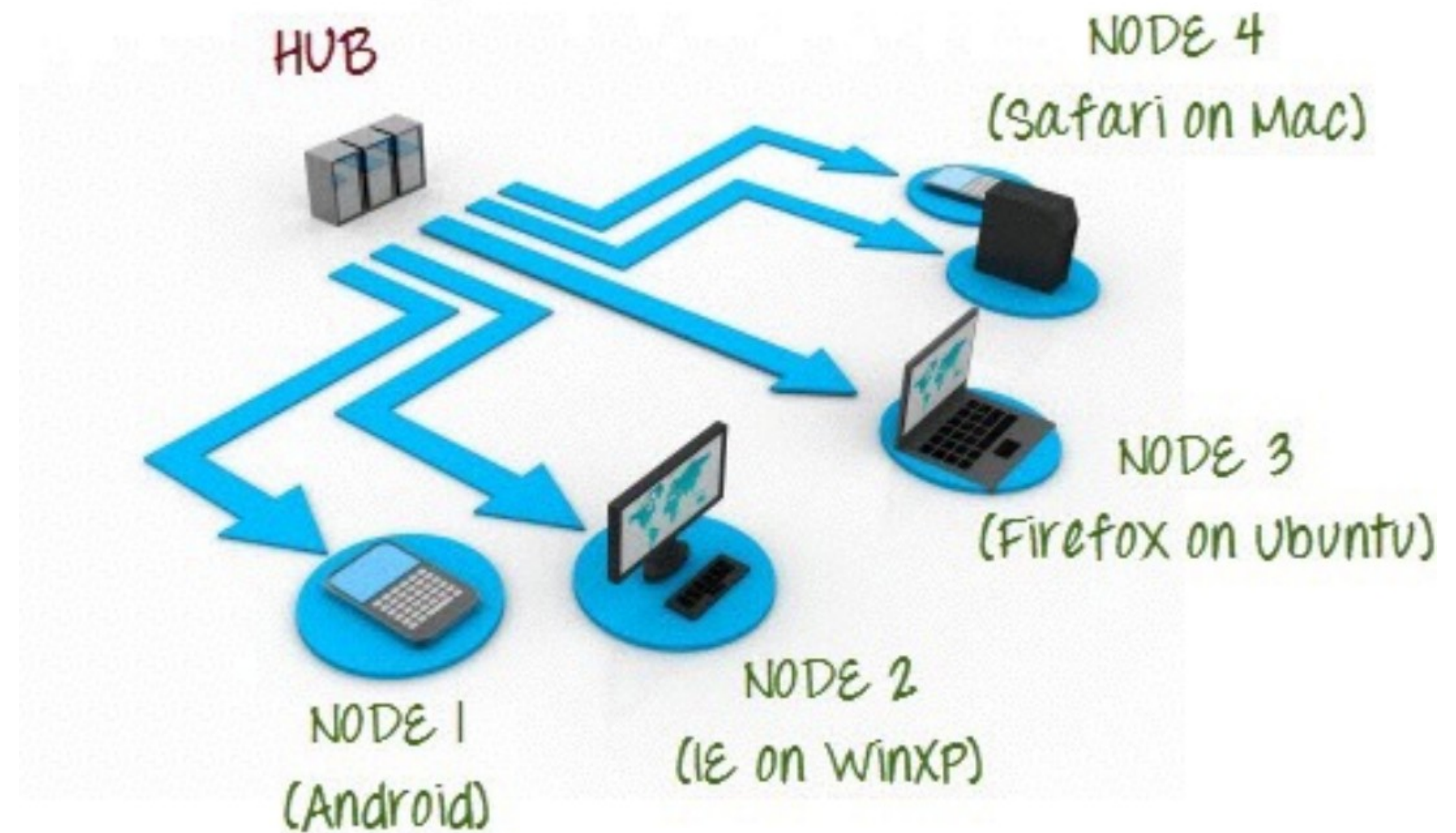


El servidor de selenium permite la ejecución de servidores que permite la ejecución de pruebas remotas

- Hub: unifica las peticiones de pruebas
- Node: Ejecuta las pruebas que manda el HUB

Selenium Grid

¿Cómo lanzo en paralelo?



Selenium Grid

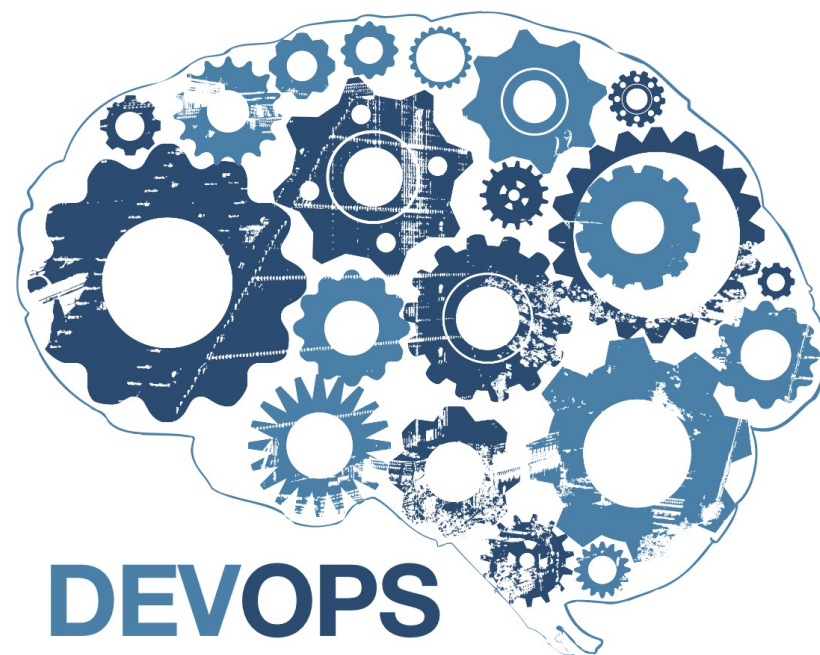
¿Cómo lanzo en paralelo?



```
// We could use any driver for our tests...
DesiredCapabilities capabilities = new DesiredCapabilities();
// ... but only if it supports javascript
capabilities.setJavascriptEnabled(true);
// Get a handle to the driver. This will throw an exception
// if a matching driver cannot be located
WebDriver driver = new RemoteWebDriver(capabilities);
// Query the driver to find out more information
Capabilities actualCapabilities = ((RemoteWebDriver)
driver).getCapabilities();
// And now use it
driver.get("http://www.google.com");
```


Conclusiones

¿Qué podemos sacar en claro?



- Hemos grabado pruebas
- Hemos exportado pruebas a Java
- Hemos cambiado las pruebas
- Hemos ejecutado las pruebas en remoto y paralelo

Referencias

¿Fuentes de información?



Cosecha propia :)

CURSOS DE DESARROLLO
DAVID VAQUERO
LICENCIA CC-BY-SA-NC 4.0

info@cursosdedesarrollo.com

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

<http://cursosdedesarrollo.com>