

SELENIUM 1:1

2 DE MAYO DE 2018



open
south
code

Vicente González
José María Gutiérrez


```
assertTrue(driver.findVayaParDeElementos(By.id("dosLocos")).arePresent().here());
```



Vicente González
@viarellano

José María Gutiérrez
@J_M_Gutierrez

Selenium

COMPONENTES, ROLES Y
RESPONSABILIDADES

COMPONENTES ROLES Y RESPONSABILIDADES

Selenium WebDriver

API - el facilitador para desarrollar el código que implementa la interacción del usuario con el navegador – *dependencia Maven*

Driver - La implementación específica de la API para cada navegador – *ejecutable que interactúa con el navegador*

Selenium IDE

Herramienta que permite generar automáticamente el código de la interacción con el navegador durante una sesión en vivo – *plugin de Firefox*

Selenium Server

El servidor que ejecuta las pruebas. Puede tener dos roles. HUB, NODE.

Hub – Servidor central donde se registran los nodos, se encarga de distribuir las pruebas entre los navegadores registrados – *selenium-server.jar*

Node – Servidor que toma el rol de un determinado navegador y se encarga de interactuar con el navegador a través del WebDriver – *selenium-server.jar*

Selenium Testing

MODELO DE TRABAJO EN
ENTORNO DE **PRUEBA**
AUTOMÁTICA Y **PRUEBA**
CONTINUA

PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN
UN ENTORNO DE INTEGRACIÓN CONTINUA

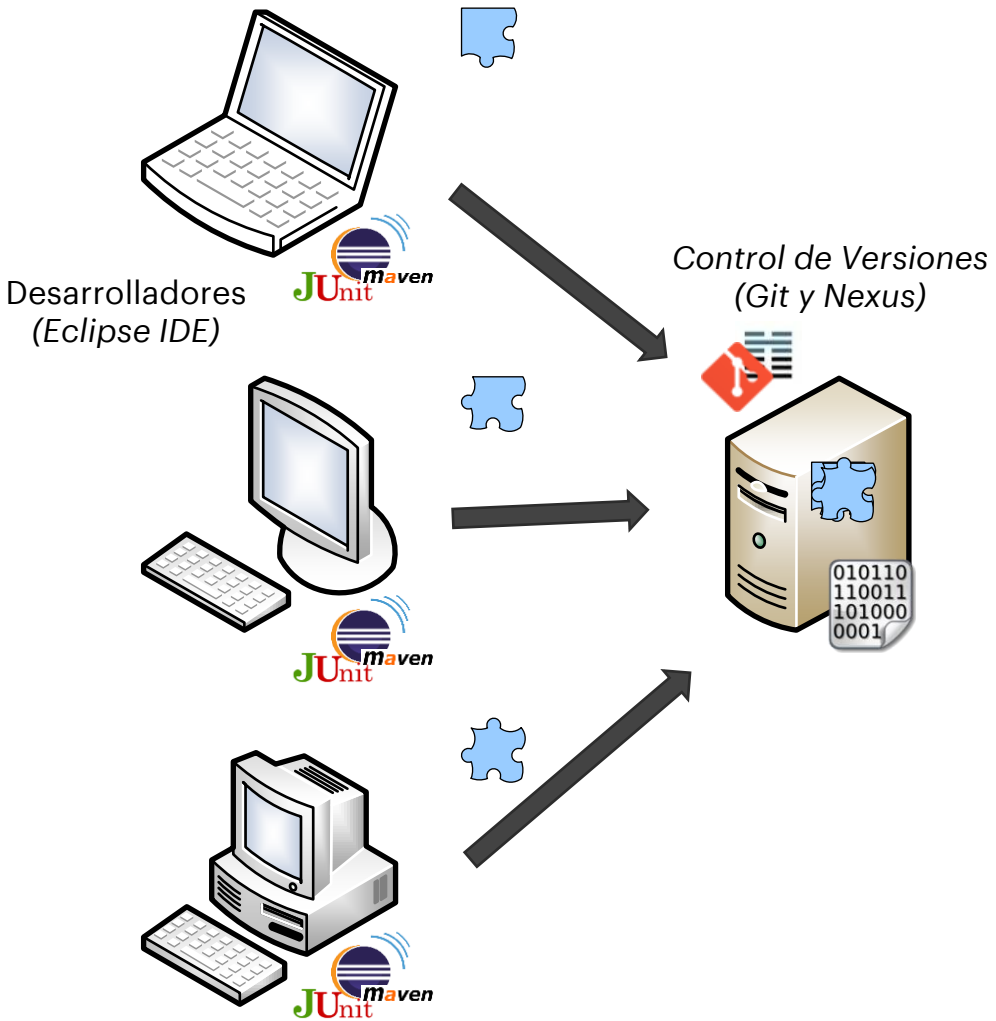


Desarrolladores
(Eclipse IDE)



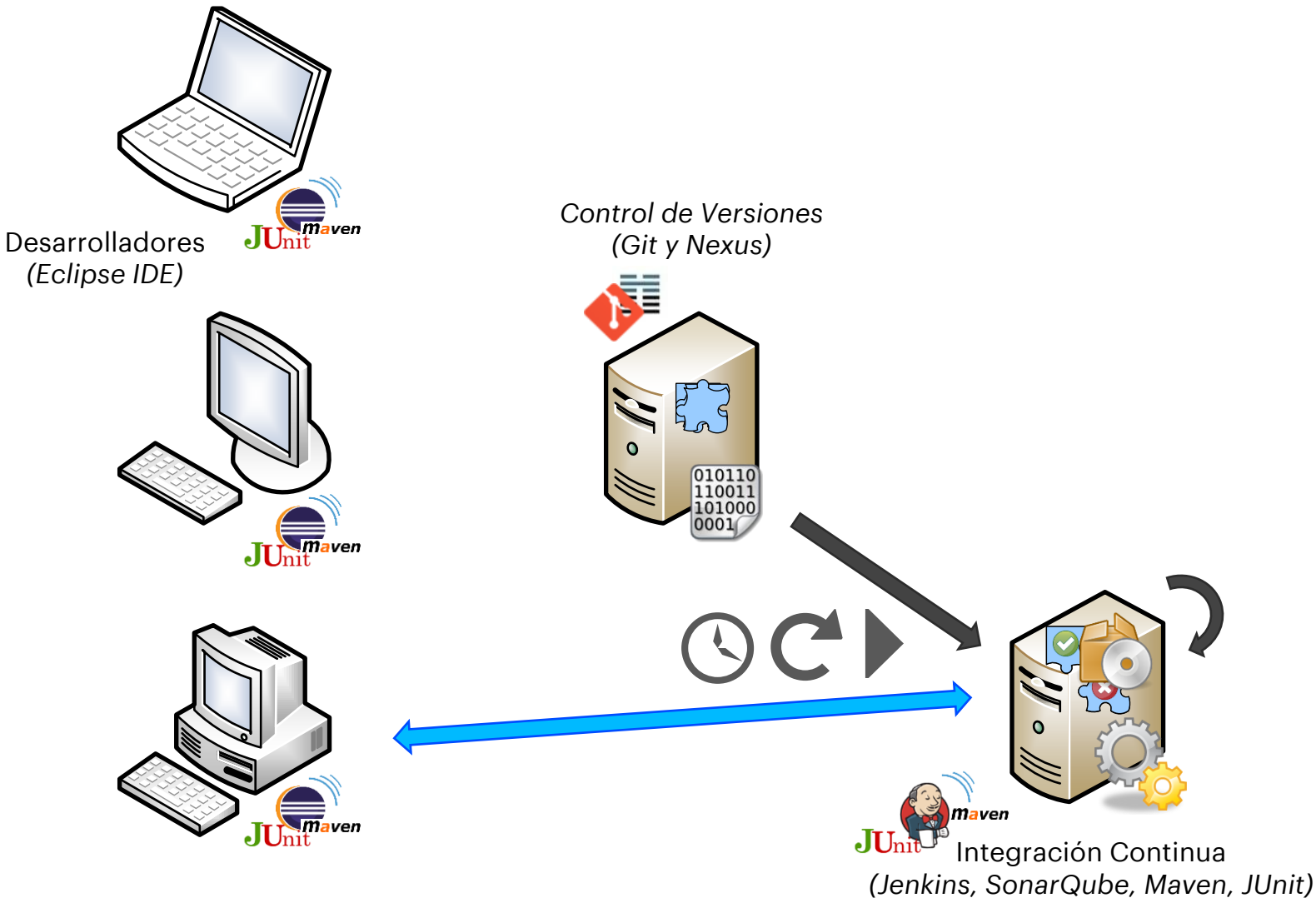
PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN
UN ENTORNO DE INTEGRACIÓN CONTINUA



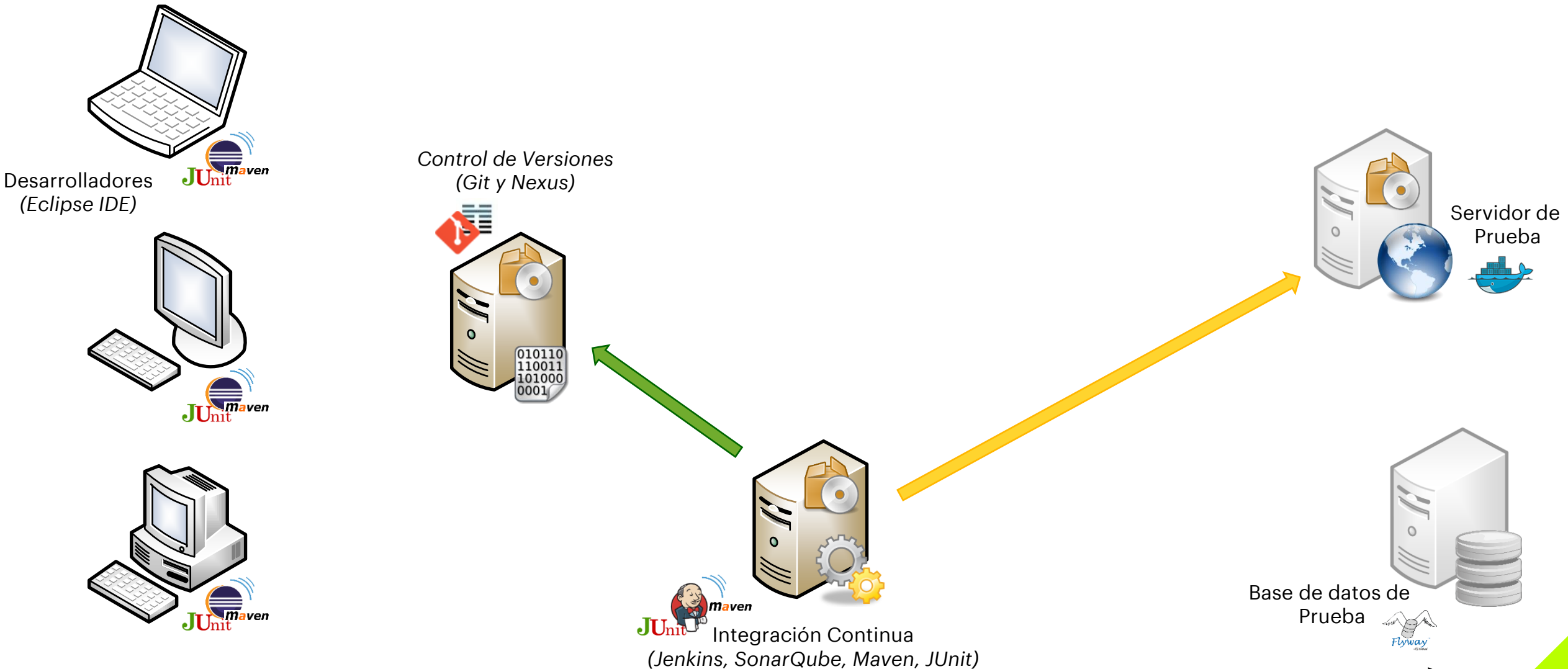
PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN
UN ENTORNO DE INTEGRACIÓN CONTINUA



PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN UN ENTORNO DE INTEGRACIÓN CONTINUA



PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN UN ENTORNO DE INTEGRACIÓN CONTINUA



Dispositivos de
Prueba



Desarrolladores
(Eclipse IDE)



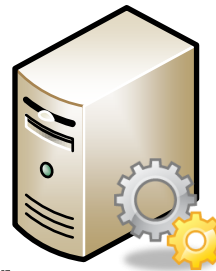
Control de Versiones
(Git y Nexus)



Test Grid Hub
(Selenium)



Servidor de
Prueba



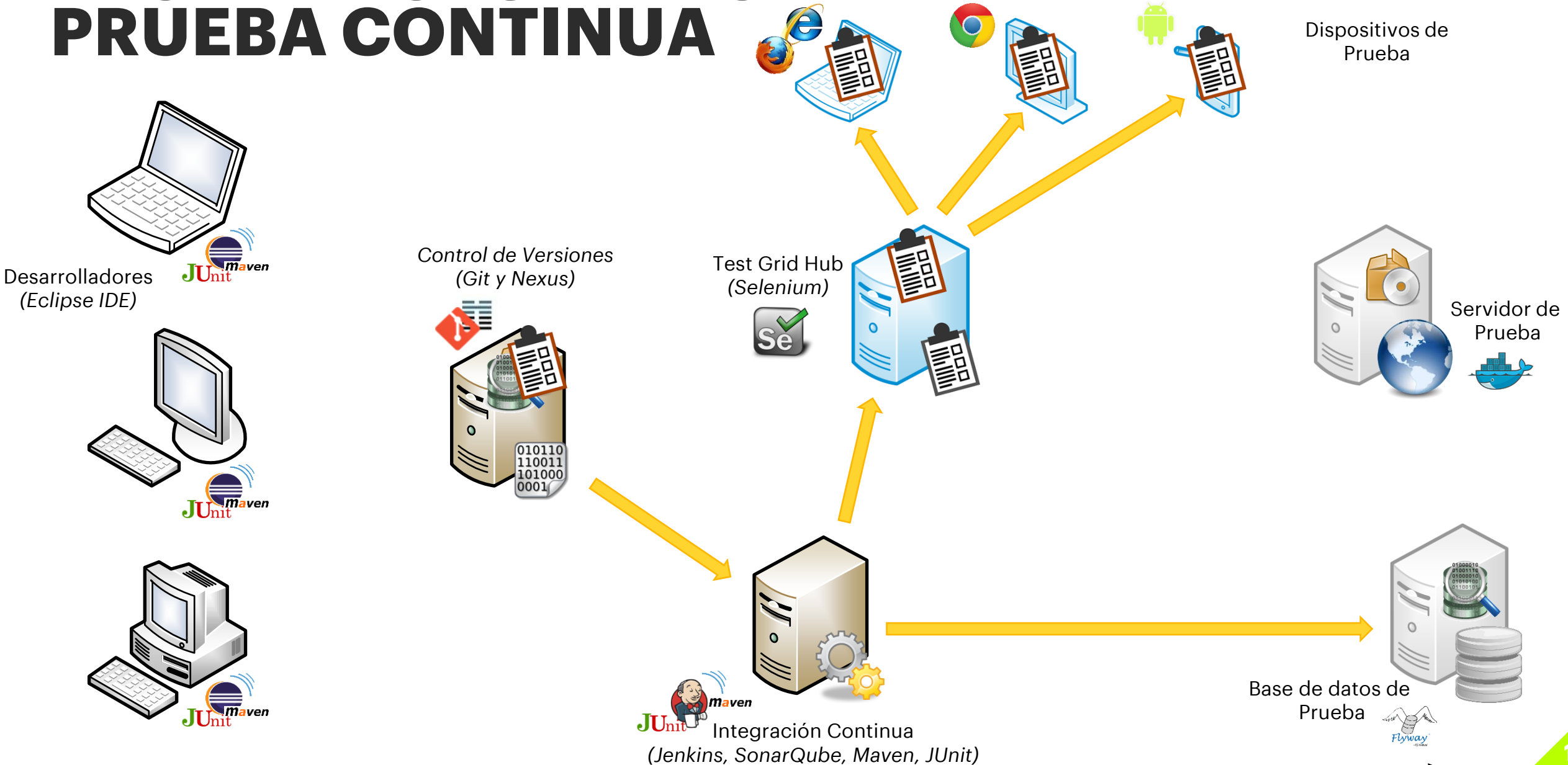
Integración Continua
(Jenkins, SonarQube, Maven, JUnit)

Base de datos de
Prueba



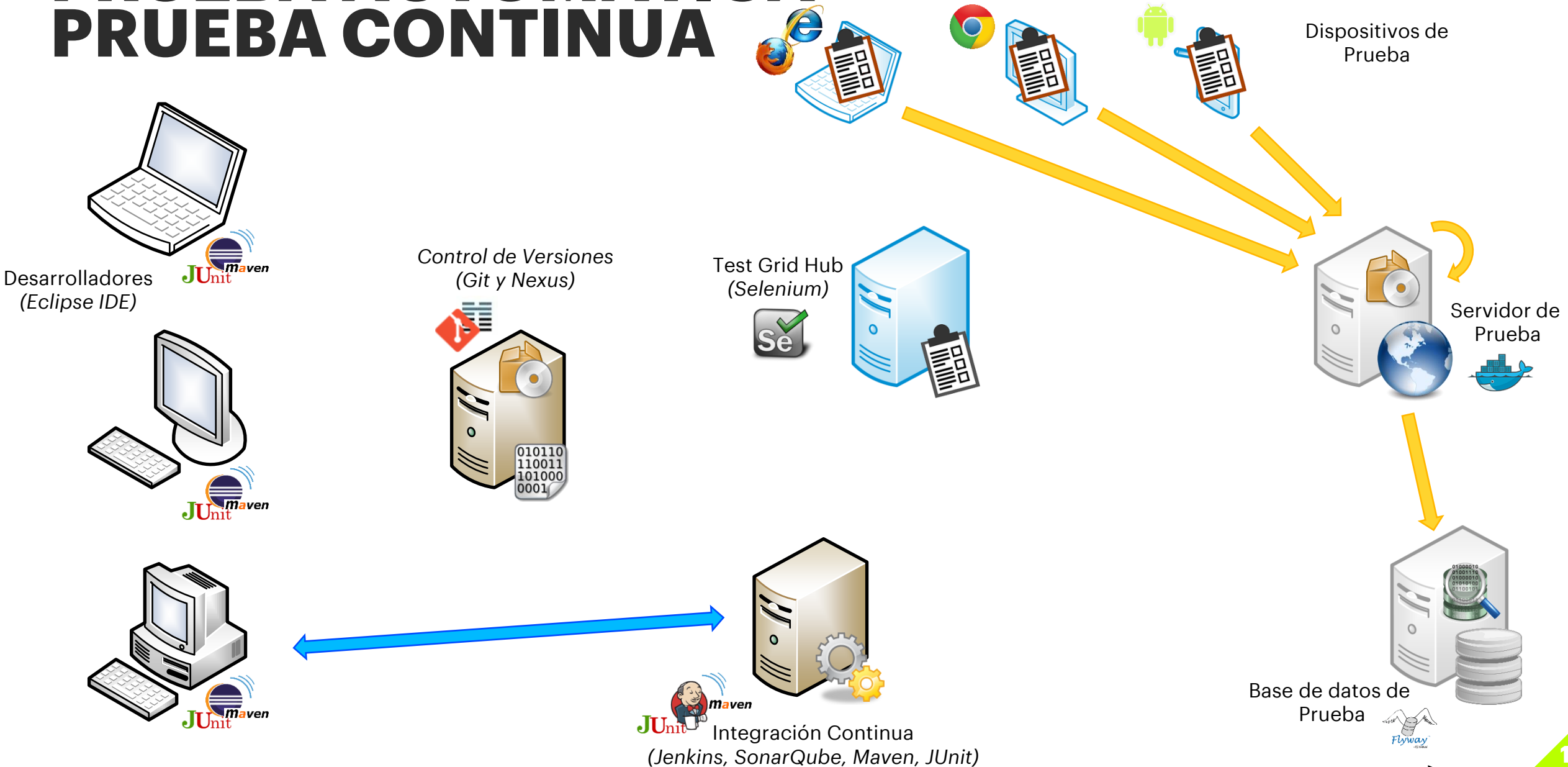
PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN UN ENTORNO DE INTEGRACIÓN CONTINUA



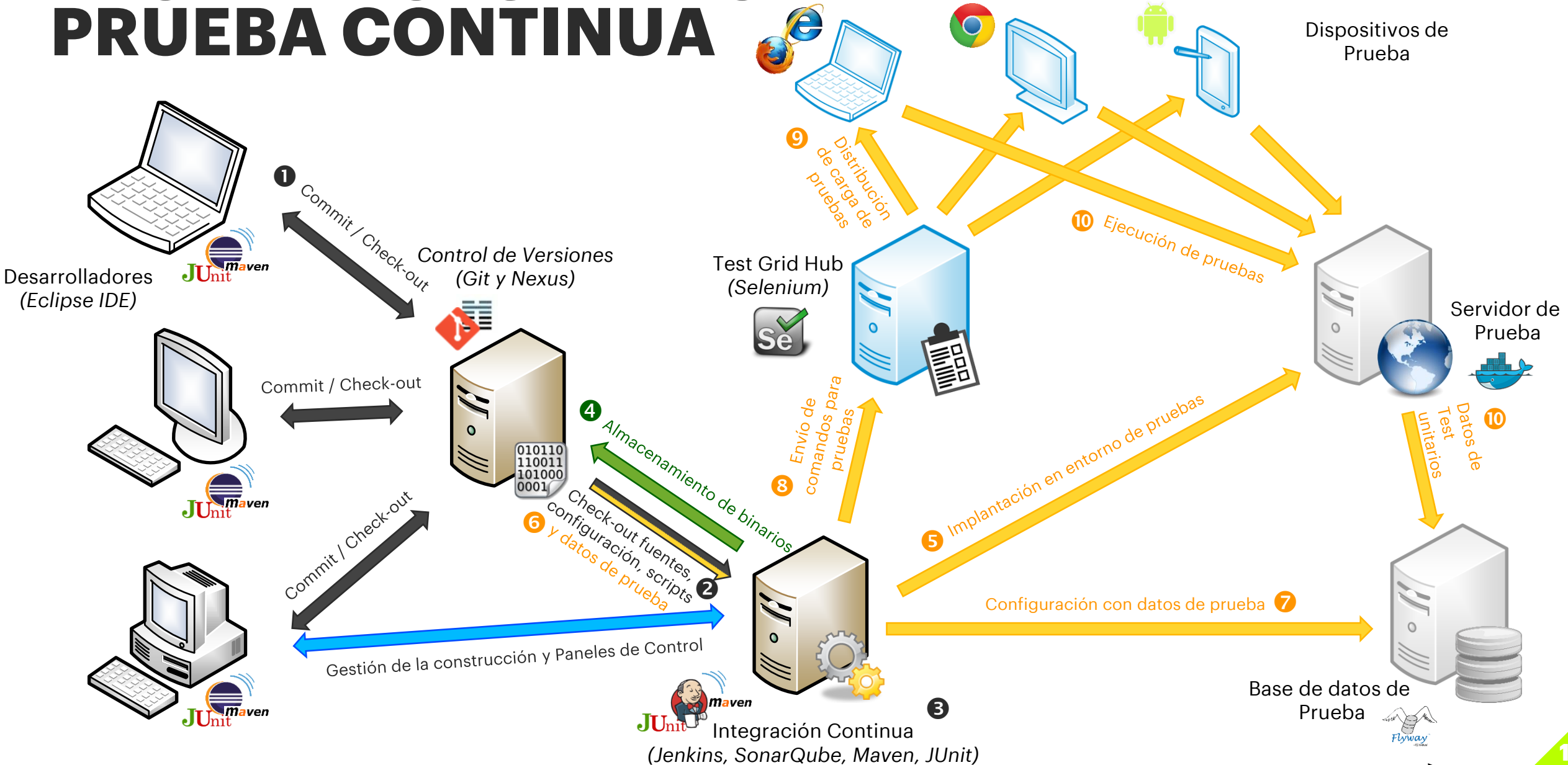
PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN UN ENTORNO DE INTEGRACIÓN CONTINUA



PRUEBA AUTOMÁTICA PRUEBA CONTINUA

MODELO DE EJECUCIÓN DE PRUEBAS AUTOMÁTICAS EN UN ENTORNO DE INTEGRACIÓN CONTINUA

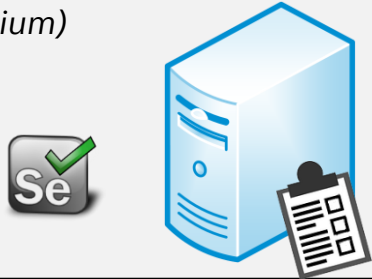


Selenium Testing

INSTALACIÓN Y
CONFIGURACION DEL
GRID DE SELENIUM

SETUP DE UN GRID

Test Grid Hub (Selenium)



```
@echo off

call ..\selenium-base\selenium-path.bat
set SELENIUM_JAR=%SELENIUM_HOME%\selenium-server-standalone-%SELENIUM_VER%.jar

start "Selenium Hub Local" java -jar %SELENIUM_JAR%
    -role hub
    -hubConfig selenium-hub-local-config.json
```

```
{ "_comment": "Configuration for Hub - selenium-hub-local-config.json",
  "maxSessions": 5,
  "port": 4444,
  "cleanupCycle": 5000,
  "timeout": 1800,
  "newSessionWaitTimeout": -1,
  "servlets": [],
  "prioritizer": null,
  "capabilityMatcher": "org.openqa.grid.internal.utils.DefaultCapabilityMatcher",
  "throwOnCapabilityNotPresent": true,
  "nodePolling": 180000,
  "debug": true,
  "platform": "WINDOWS" }
```

Dispositivos de Prueba



```
@echo off

call ..\selenium-base\selenium-path.bat
set SELENIUM_JAR=%SELENIUM_HOME%\selenium-server-standalone-%SELENIUM_VER%.jar

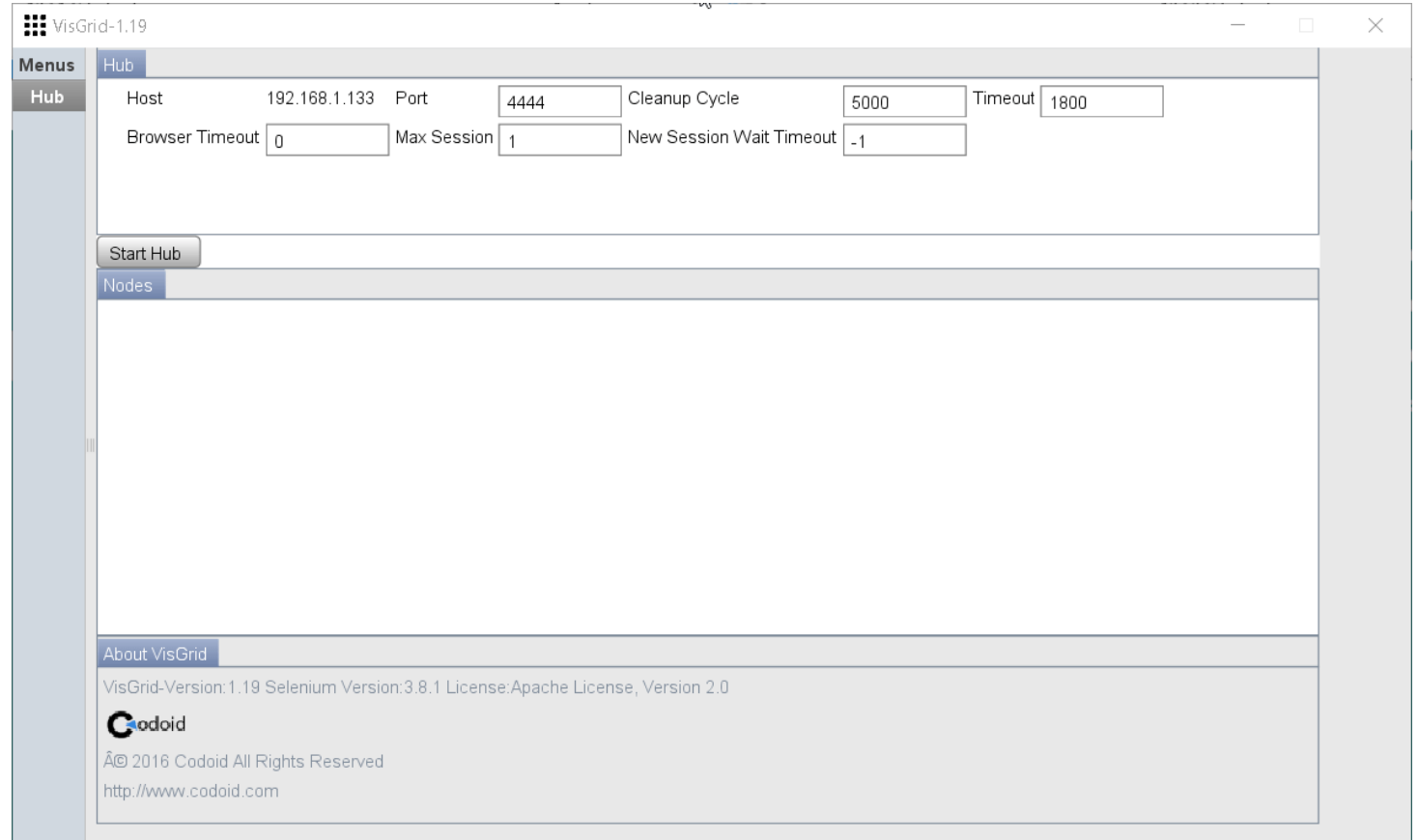
start "Selenium Edge Node to Local Hub" java -jar %SELENIUM_JAR%
    -role node
    -nodeConfig selenium-node_hub_local_edge.json
```

```
{
  "capabilities":
  [
    {
      "browserName": "MicrosoftEdge",
      "maxInstances": 5,
      "webdriver.edge.driver": "C:/java/qa/selenium-driver/edge/MicrosoftWebDriver.exe"
    }
  ],
  "proxy": "org.openqa.grid.selenium.proxy.DefaultRemoteProxy",
  "maxSession": 5,
  "port": 5556,
  "register": true,
  "registerCycle": 5000,
  "hub": "http://localhost:4444",
  "nodeStatusCheckTimeout": 5000,
  "nodePolling": 5000,
  "role": "node",
  "unregisterIfStillDownAfter": 60000,
  "downPollingLimit": 2,
  "debug": false,
  "servlets": [],
  "withoutServlets": [],
  "custom": {}
}
```

#TIP: USAR **VISGRID** PARA MONTAR EL GRID

VISGRID

**APLICACIÓN JAVA DE
ESCRITORIO QUE
FACILITA LA
CONFIGURACIÓN DE
UN GRID**



Usar en entornos locales de desarrollo, no productivos

VisGrid is provided by Codoid (<https://codoid.com/>) under Apache License

#TIP: CREANDO GRIDS EFÍMEROS

Usando GridLauncher para crear instancias de Selenium Server con diferentes roles

```
<dependency>
  <groupId>org.seleniumhq.selenium</groupId>
  <artifactId>selenium-server</artifactId>
  <version>${selenium.version}</version>
</dependency>
```

```
// Start hub
GridLauncherV3.main(new String[] { "-role", "hub", "-port", "4449", "-log", "grid.out" });
```

```
// Register Chrome in hub
System.setProperty("webdriver.chrome.driver", "C:/java/qa/selenium-driver/chromedriver.exe");
logger.info("\t-- --webdriver.chrome.driver: " + System.getProperty("webdriver.chrome.driver"));
GridLauncherV3.main(new String[] { "-role", "node", "-hub", "http://localhost:4449/grid/register", "-browser",
    "browserName=chrome", "-port", "5558", "-log", "chrome.out" });
```

#TIP: USO DEL WEBDRIVERMANAGER PARA GESTIONAR LOS BINARIOS

WebDriverManager – Un proyecto Open Source que permite realizar un control explícito de los binarios de Selenium WebDriver

<https://github.com/bonigarcia/webdrivermanager>

```
<dependency>
  <groupId>io.github.bonigarcia</groupId>
  <artifactId>webdrivermanager</artifactId>
  <version>1.7.2</version>
  <scope>test</scope>
</dependency>
```

```
public static void setUpNodeDrivers() throws Exception {
    logger.info("\n");
    logger.info("##### Setting up node drivers..." + WEBDRIVER_SETUP);

    ChromeDriverManager.getInstance().setup();
    logger.info("\t-- --webdriver.chrome.driver: " + System.getProperty("webdriver.chrome.driver"));

    FirefoxDriverManager.getInstance().setup();
    logger.info("\t-- --webdriver.gecko.driver: " + System.getProperty("webdriver.gecko.driver"));
}
```


#TIP: USO DEL WEBDRIVERMANAGER PARA GESTIONAR LOS BINARIOS

Configurable usando el API

Example	Description
<code>ChromeDriverManager.getInstance().version("2.26").setup();</code>	Force to use version 2.26 of <i>chromedriver</i>
<code>FirefoxDriverManager.getInstance().arch32().setup();</code>	Force to use the 32-bit version of <i>geckodriver</i>
<code>OperaDriverManager.getInstance().forceCache().setup();</code>	Force to use the cache version of <i>operadriver</i>
<code>PhantomJsDriverManager.getInstance().useTaobaoMirror().setup();</code>	Force to use the taobao.org mirror to download <i>phantomjs</i> driver
<code>ChromeDriverManager.getInstance().proxy("server:port").setup();</code>	Using proxy <i>server:port</i> for the connection

O a través de parámetros de configuración

```
wdm.targetPath=~/.m2/repository/webdriver
wdm.forceCache=false
wdm.override=false
wdm.timeout=30
wdm.seekErrorRetries=3

wdm.chromeDriverUrl=https://chromedriver.storage.googleapis.com/
wdm.chromeDriverTaobaoUrl=http://npm.taobao.org/mirrors/chromedriver
wdm.chromeDriverExport=webdriver.chrome.driver
wdm.chromeDriverVersion=LATEST
```

A high-angle photograph of a person's lower half and legs, wearing a grey textured coat and black pants, pulling a bright yellow rolling suitcase on a grey stone-paved surface. The person is walking away from the camera towards the top left.

para llevar:
**Elige tu propia
estrategia para
montar el grid**

**Un grid de Selenium
debe estar en un
entorno separado y
accessible del entorno
desde donde se
ejecuten las pruebas.
Reusa, Estandariza.**

Selenium Testing

DESARROLLANDO LOS
SCRIPTS DE PRUEBAS

#TIP: EVITA EL USO DE **SELENIUM IDE**

La generación automática del código de las pruebas va en contra de las buenas prácticas de:

Estandarización

Reutilización

Uso de patrones

El código de pruebas es siempre el más sencillo, basado en patrones básicos:

→ **dato de la prueba**

→ **ejecución de la aplicación objeto de la prueba**

→ **verificación resultado esperado**

#TIP: USO DEL **PATRON PAGE OBJECT**

El patrón **Page Object** trata de aislar los scripts de pruebas de la propia interacción con el navegador.

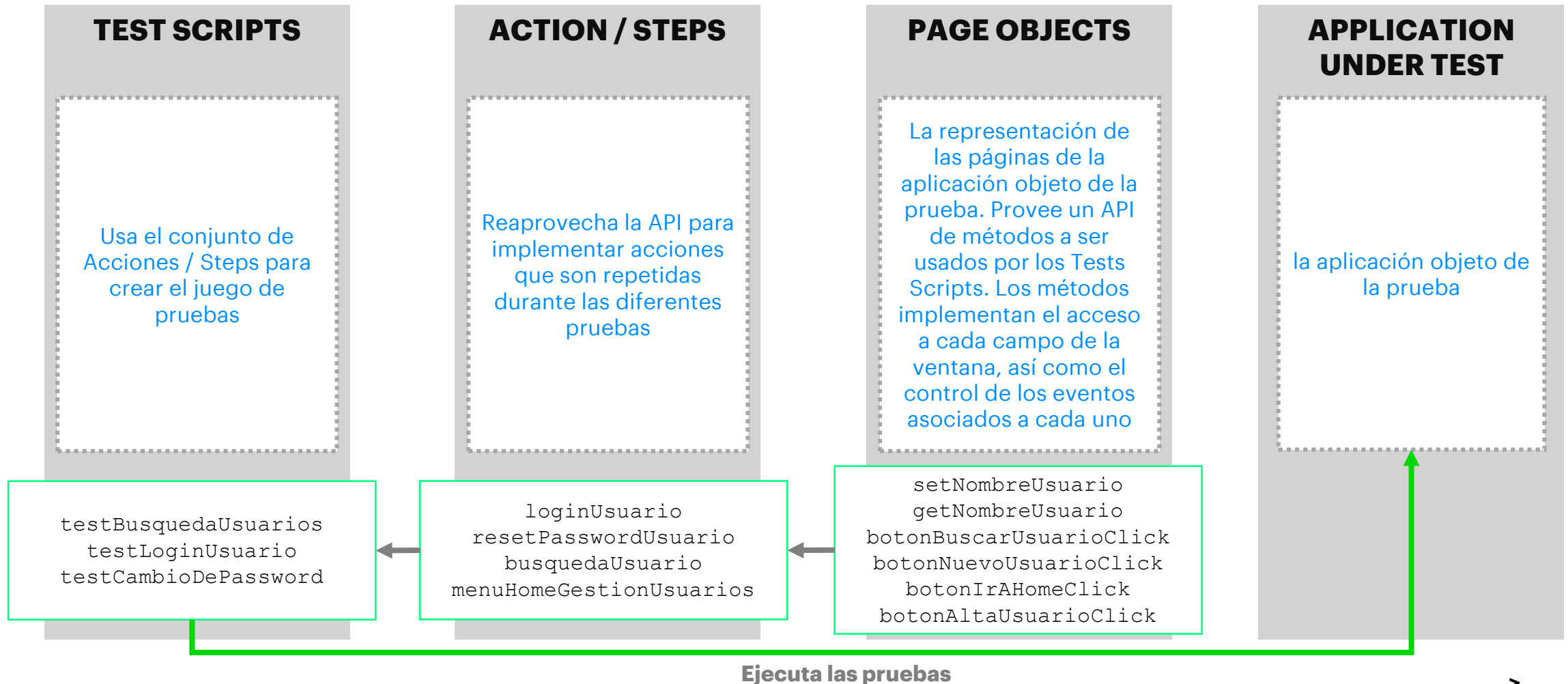
Consiste en la creación de un **Page Object** por cada **página** de la aplicación a ser probada.

El Page Object se encargará de implementar cada interacción del usuario con la aplicación (acceso a cada campo de la ventana, ejecución de cada evento, recarga de componentes, etc).

La capa de Page Objects provee de un **API** para la creación de las pruebas.

#TIP: USO DEL **PATRON PAGE OBJECT**

Diagrama referencial de desarrollo de test scripts de Selenium sobre el patrón Page Object



#TIP: **XPATH** NO ESTA BAJO TU **CONTROL**

El uso de Xpath en Selenium depende de:

Las capacidades nativas de cada navegador

En aquellos casos que no esté disponible, es el webdriver el que provee su propia implementación.

How XPath Works in WebDriver

At a high level, WebDriver uses a browser's native XPath capabilities wherever possible. On those browsers that don't have native XPath support, we have provided our own implementation. This can lead to some unexpected behaviour unless you are aware of the differences in the various xpath engines.

Driver	**Tag and Attribute Names**	**Attribute Values**	**Native XPath Support**
:-----	:-----	:-----	:-----
HtmlUnitDriver	Lower-cased	As they appear in the HTML	Yes
InternetExplorerDriver	Lower-cased	As they appear in the HTML	No
FirefoxDriver	Case insensitive	As they appear in the HTML	Yes

#TIP: **XPATH** NO ESTA BAJO TU **CONTROL**

Ejemplo ilustrativo:

This is a little abstract, so for the following piece of HTML:

```
...  
<input type="text" name="example" />  
<INPUT type="text" name="other" />  
...
```

The following will happen:

XPath expression	**Number of Matches In**		
:-----	:-----	:-----	:-----
	HtmlUnitDriver	**FirefoxDriver**	**InternetExplorerDriver**
//input	1 ("example")	2	2
//INPUT	0	2	0

#TIP: ALTERNATIVE A XPATH: ASIGNAR ID PREDECIBLES A TU APLICACIÓN

En lugar de crear vías **complejas** para acceder a los elementos en la aplicación web, considera **cambiar** tu aplicación:

- **Sobreescribir** ID de los components de la página
- Crear anclajes intermedio que **faciliten** el acceso a los elementos

A high-angle photograph of a person's lower half and legs, wearing a grey textured coat, black trousers, and black shoes. They are pulling a bright yellow rolling suitcase on a light grey paved surface. The suitcase is upright and has a black handle and straps.

para llevar:
**Pensar en la
testabilidad**

**Uno de los drivers que
debe premiar a la hora
de escoger tu
framework web, es la
capacidad que ofrezca
de ser probado.
TESTABILIDAD**

credits

Repositorios:

<https://github.com/viarellano/deors.demos.petclinic>

<https://github.com/deors/deors.demos.petclinic>