

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

Estas en: Inicio Tutoriales Ajax tests con Selenium: prototype.js e ICEfaces.

Últimas Noticias

- » Historia de la Informática. Capítulo 66. 1988
- » Historia de la informática. Capítulo 65. 1987
- » Autentia en la Sun Open Communities Forum
- » Comentario del libro: El economista naturalista de Robert Frank
- » Contratos ágiles: Vendiendo Scrum a tus clientes.
- » Resumen de la cuarta charla gratuita de Autentia: SCRUM (con video)
- » Si se pregunta ¿Qué ofrece este Web?
- » Vota AdictosAltrabajo.com en DZone
- » Autentia cumple 6 años

+Noticias Destacadas

- » Contratos ágiles: Vendiendo Scrum a tus clientes.
- » Quinta charla Autentia + Projectails + Agile Spain: Contratos ágiles: Vendiendo Scrum a tus clientes
- » Lo mejor de esta semana: Curso de Scrum con Ángel Medinilla
- » Resumen de la cuarta charla gratuita de Autentia: SCRUM (con video)

+Comentarios Cómico

+Enlaces

Tutorial desarrollado por



Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos. Diseñador de Adictos Al Trabajo 2.0

Puedes encontrarme en [Autentia](#)

Somos expertos en Java/J2EE

Catálogo de servicios de Autentia

Descargar (6,2 MB)

Descargar en versión comic (17 MB)

AdictosAltrabajo.com es el Web de difusión de conocimiento de Autentia.



Catálogo de cursos

Descargar este documento en formato PDF: [seleniumAjaxTestingPrototypejsICEfaces.pdf](#)

Fecha de creación del tutorial: 2009-07-13

Ajax tests con Selenium: prototype.js e ICEfaces.

0. Índice de contenidos.

1. Introducción.
2. Entorno.
3. waitForCondition.
4. Una solución mejor: basada en prototype.js.
5. Una solución mucho mejor: waitForAjaxRequest.
6. Testing ICEfaces partialSubmit.
7. Conclusiones.

1. Introducción

En este tutorial vamos a hablar de cómo escribir tests funcionales con Selenium IDE sobre aplicaciones que realizan recargas controladas de la interfaz de usuario con Ajax.

Si no estás familiarizados con la tecnología, podéis encontrar, [enadictos](#), tutoriales sobre Selenium y Ajax. En la redacción de este tutorial se da por hecho que el lector tiene experiencia con ambas, además de conocimientos de [prototype.js](#) e [ICEfaces](#).

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Asus G1 (Core 2 Duo a 2.1 GHz, 2048 MB RAM, 120 GB HD).
- Sistema operativo: Windows Vista Ultimate.
- Firefox 3.5, con Selenium IDE 1.0.2.
- Prototype.js 1.6.0.3.
- ICEfaces 1.8.0.

3. waitForCondition.

No existe, en la fecha de publicación de este tutorial, un comando específico de Selenium para esperar la respuesta a una petición Ajax. Para ello se usa el comando `waitForCondition` que recibe como parámetro una expresión javascript que será evaluada, durante un tiempo determinado, hasta que devuelva "true".

Dicha expresión:

- acepta más de una línea, si bien solo será evaluado el resultado de la última.
- no se evalúa en el entorno de la ventana de la aplicación, con lo que, para acceder a objetos de nuestra ventana, debemos utilizar `selenium.browserbot.getCurrentWindow()` y, si queremos acceder al javascript de la página, mejor `selenium.browserbot.getUserWindow()`.

Vamos a tomar como referencia el ejemplo que se plantea en el [tutorial de recarga controlada de selectores con prototype](#) para llevar a cabo un test de su funcionalidad. La interfaz de usuario presenta dos selectores, un evento de cambio de valor sobre el primero provoca una recarga del contenido del segundo, y dicha recarga se realiza en segundo plano, sin recargar la página completa, a través del soporte para Ajax que nos proporciona [prototype.js](#).

En el ejemplo la petición se realiza de forma síncrona, con lo que no sería necesario ningún comando de espera, puesto que el navegador ya se encarga de esperar la respuesta del servidor, con lo que, a efectos de realizar la prueba, debemos modificar el parámetro que asigna el tipo de conexión, indicado que la misma será asíncrona.

```

01. // prototype Ajax.Request: petición de servicio
02. var response = new Ajax.Request(sur1, {asynchronous: true, method: 'post', parameters: params}).transport;
    
```

La función que realiza la recarga, antes de enviar la petición, limpia el segundo selector, dejando su número de options a cero. No aprovecharemos de dicha "feature" para conocer cuándo se ha producido la respuesta, cuando el número de options del select sea distinto mayor que cero se ha habra producido la misma.

La siguiente secuencia de comandos muestra cómo se podría realizar un test de dicha funcionalidad:

Command	Target	Value
open	http://localhost:8080/prototypeAjax/	
assertTitle	Recarga de selectores	
select	provincias	label=Madrid
waitForCondition	selenium.browserbot.getCurrentWindow().document.getElementById('poblaciones').options.length > 0	5000
select	poblaciones	label=Alcobendas

El comando `waitForCondition` accede al selector de poblaciones, para comprobar su contenido, esto es, el número de options. Cuando se produce la respuesta Ajax, dicho selector se pobla con el contenido de la misma y deja de tener una longitud de cero.

El segundo parámetro del comando (columna de Value) es el tiempo de timeout máximo, durante el cual se repetirá la evaluación de la condición hasta que:

- la expresión devuelva "true", en cuyo caso, se continuará con el siguiente comando,
- expire el tiempo de timeout, terminando la ejecución del TestCase con un error.

Esta solución es perfectamente válida, si bien, requiere de un análisis individual de cada uno de los eventos que producen una recarga, vía Ajax, en nuestra aplicación, para buscar una condición javascript que pueda ser usada y evaluar cuando se reproduce la respuesta.

4. Una solución mejor: basada en prototype.js.

Vamos a intentar evitar la necesidad de buscar una condición individual que evalúe cuando se produce la respuesta, de modo que no tengamos que ir caso por caso, analizando los eventos Ajax que se producen en nuestra aplicación.

La mejor manera de llevarlo a cabo es bajar al nivel del framework que maneja las conexiones, en nuestro caso [prototype.js](#) y lo podemos hacer de dos maneras distintas:

- enganchándonos al ciclo de vida de la petición Ajax, de modo que programemos una función de callback que sea invocada al inicio y final de cada petición, (podéis ver [un ejemplo de registro via onComplete: en este post](#)) o
- accediendo a la variable que registra el número de conexiones activas.

En este caso, vamos a optar por la segunda de las opciones por su sencillez, y, sobre la base del mismo ejemplo podríamos programar la siguiente secuencia de comandos:

Command	Target	Value
open	http://localhost:8080/prototypeAjax/	
assertTitle	Recarga de selectores	
select	provincias	label=Madrid
waitForCondition	selenium.browserbot.getUserWindow().Ajax.activeRequestCount == 0	5000

Catálogo de servicios Autentia (PDF 6,2MB)



En formato comic...



Web
 www.adictosaltrabajo.com

Últimos tutoriales

2009-07-13
Ajax tests con Selenium: prototype.js e ICEfaces.

2009-07-08
AOP con AspectJ y Maven

2009-07-07
Instalación y configuración de Eclipse Galileo

2009-07-07
Iniciarse en el manejo de JME, Creación de un Cloth.

2009-07-06
Primeros pasos con Blender: Pintando nuestra mascota en 3D

2009-07-06
DBUnit-Exportar e Importar BBDD

2009-07-05
JMeter, Pruebas de stress sobre aplicaciones web: Grabando y reproduciendo navegaciones

2009-07-02
Axis2: Invocación de Servicios Web usando distintos MEP

2009-07-02
Instalación OpenOffice

2009-07-02
Juegos 3D en Java: Blender y JMonkeyEngine

2009-06-20
StAX (Xml Pull Parser): Streaming API para XML

2009-06-15
Configuración de la desconexión de usuarios con ICEfaces

2009-06-10
LWUIT: Una librería gráfica tipo AWT o Swing para J2ME

2009-06-10
Mapas mentales con XMind

2009-02-26
Redimensionar Imágenes en Windows Vista

2009-06-08
UploadFile con Icefaces + Hibernate + Anotaciones

2009-06-05
Habilitar exportación en Liferay

2009-06-01
Registrar Liferay en Eclipse

2009-05-29
Liferay Social Office

2009-05-28
Broadcast con Ustream

2009-05-25
Tabla datos accesible con ordenación y paginación

2009-05-21
Primeros pasos con Audacity: Un editor de sonido libre y multiplataforma.

2009-05-11
Introducción a TortoiseSVN

2009-05-07
Hacer 'scp' de varios ficheros sin solicitud de clave

2009-05-02
Plugin Hibernate3 para Maven

Command	Target	Value
select	poblaciones	label=Alcobendas

Ahora sí, la condición es genérica y, como tal, susceptible de incorporarse a un comando personalizado que pueda ser invocado en cualquiera de nuestros scripts de Selenium, como si de un comando propio del IDE se tratase.

5. Una solución mucho mejor: waitForAjaxRequest.

Selenium es plugable, esto es, ofrece la posibilidad de incluir nuestros propios scripts con comandos personalizados.

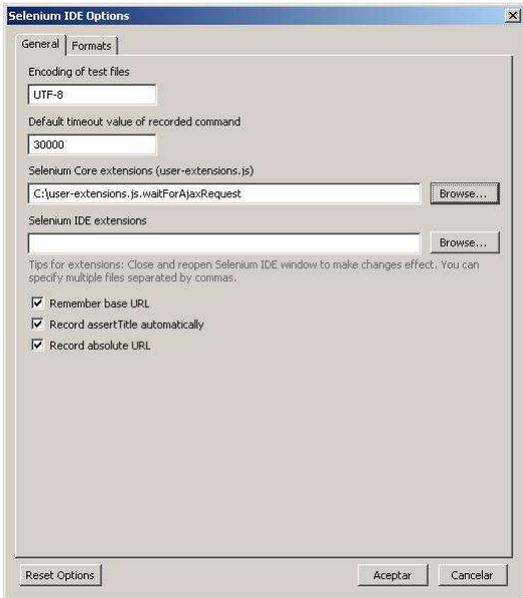
Basándonos en la condición anterior podemos crear la siguiente función:

```

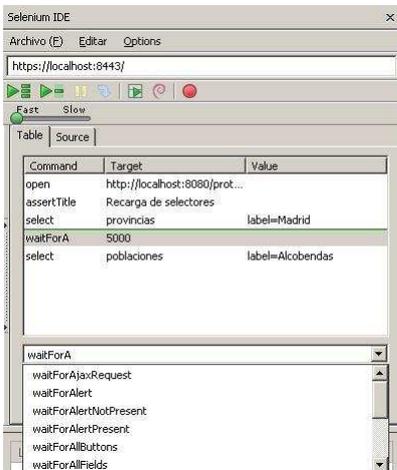
view plain print ?
01.  /**
02.   * Based in prototype.js framework, this routine will wait until
03.   * an Ajax request has finished and then return.
04.   */
05.  Selenium.prototype.waitForAjaxRequest = function(timeout) {
06.    return Selenium.decorateFunctionWithTimeout(function() {
07.      if (selenium.browserbot.getUserWindow().Ajax.activeRequestCount ==0) {
08.        return true;
09.      }
10.      return false;
11.    }, timeout);
12.  }

```

Para poder hacer uso de ella basta con incluirla en un fichero .js y añadirlo al entorno del IDE, en el menú de opciones:



Será necesario cerrar y volver a abrir la ventana del IDE, para que lea el fichero, y ya tendremos nuestro comando dentro del catálogo de disponibles.



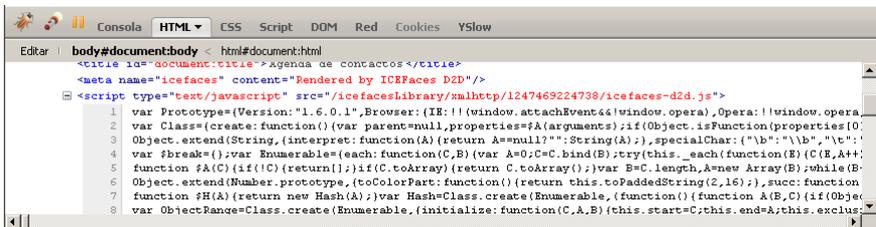
Siguiendo con el mismo ejemplo, la secuencia de comandos quedaría como sigue:

Command	Target	Value
open	http://localhost:8080/prototipeAjax/	
assertTitle	Recarga de selectores	
select	provincias	label=Madrid
waitForAjaxRequest	5000	
select	poblaciones	label=Alcobendas

Mucho más limpio, ni que decir tiene que si pasamos a utilizar otro framework que gestione las conexiones vía Ajax, el único punto de colisión sería nuestro script.

6. Testing ICEfaces partialSubmit.

Conociendo que ICEfaces hace uso extensivo de [prototype.js](#) para la recarga del árbol DOM en cliente (Direct-to-DOM rendering), sería de esperar que hiciesen también uso del soporte para Ajax que proporciona dicha librería, máxime cuando distribuyen el código fuente.



- 2009-04-26 [AgileDraw: una técnica rápida de modelado](#)
- 2009-04-24 [Spring AOP: Cacheando aplicaciones usando anotaciones y aspectos con Aspectj](#)
- 2009-04-20 [Modelos de conocimiento con CmapTools](#)
- 2009-04-16 [Informes Crosstab con iReport](#)
- 2009-04-16 [Registro de un fichero de datos personales con el formulario NOTA](#)
- 2009-04-15 [Estadísticas de www.adictosaltrabajo.com Abril 2009](#)
- 2009-04-15 [Iniciación a OSWorkflow con Spring](#)
- 2009-04-14 [Tests de Selenium con librerías de componentes JSF: Apache Tomahawk.](#)
- 2009-04-13 [JTAPL. El API de Telefonía para Java](#)
- 2009-04-13 [Registro de Web Services con Apache JUDDI. Configuración y ejemplo](#)
- 2009-04-13 [Cómo hacer UML con Eclipse y el plugin UML2](#)
- 2009-04-09 [Spring WS: Servicios Web a través del correo electrónico](#)
- 2009-04-02 [Creación de cursos con Moodle](#)
- 2009-03-31 [Integrar Liferay Portal 5.2.1 con Pentaho BI 2.0.0 sobre MySQL 5.1](#)
- 2009-03-31 [Spring WS: Construcción de Clientes de Servicios Web con Spring](#)
- 2009-03-30 [Administración de sitios Moodle](#)
- 2009-03-29 [Empaquetamiento de aplicaciones de escritorio \(standalone\) con Maven](#)
- 2009-03-27 [Primeros pasos con Moodle](#)
- 2009-03-26 [Introducción a JSF Java](#)
- 2009-03-25 [A1 Website Analyzer](#)
- 2009-03-24 [Cómo ver el correo de Gmail sin conexión a Internet](#)
- 2009-03-20 [JasperReports Maven Plugin](#)
- 2009-03-16 [Creación de contenidos SCORM: eXe](#)
- 2009-03-15 [Spring WS: Creación de Servicios Web con Spring](#)
- 2009-03-13 [Instalación Alfresco \(Labs\)](#)
- 2009-06-29 [Atención a cliente - Call Center - BARCELONA.](#)
- 2009-06-25 [Atención a cliente - Call Center - BARCELONA.](#)
- 2009-06-20 [Comercial - Ventas - CASTELLON.](#)
- 2009-06-19 [Otras - Ingeniería \(minas, puentes y puertos\) - VALENCIA.](#)
- 2009-06-17 [Comercial - Ventas - ALICANTE.](#)
- [Anuncios Google](#)

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com



Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
AOP con AspectJ y Maven	Programacion orientada a aspectos con AspectJ y Maven	2009-07-08	224	Bueno	1	
Configuración de la desconexión de usuarios con ICEFaces	Este tutorial muestra la manera de configurar y traducir la ventana de desconexión o pérdida de sesión del usuario en ICEFaces.	2009-06-15	840	Muy bueno	7	
UploadFile con Icefaces + Hibernate + Anotaciones	Este tutorial pretende servir de guía de inicio para todos aquellos usuarios que comienzan a dar sus primeros pasos en ICEfaces, JSF e Hibernate	2009-06-08	1053	Bueno	14	
Tests de Selenium con librerías de componentes JSF: Apache Tomahawk.	En este tutorial vamos a hablar de cómo escribir tests funcionales con Selenium IDE sobre interfaces de usuario construidas con librerías de componentes visuales JSF y, en concreto, con Apache Tomahawk y uno de sus componentes.	2009-04-14	1603	Muy bueno	4	
Creación de cursos con Moodle	En este tercer tutorial de la serie veremos en detalle la creación de un curso	2009-04-02	2902	Muy bueno	11	
Introducción a JSF Java	Hablaremos del funcionamiento de JSF, las páginas JSF, las etiquetas JSF, los managed beans, la navegación entre páginas, el ciclo de vida de la petición al servidor y otros temas.	2009-03-26	6687	Bueno	39	
Cómo ver el correo de Gmail sin conexión a Internet	En este tutorial vamos a ver como podemos configurar el navegador Firefox 3 para poder acceder a todo nuestro histórico de correos sin necesitar de un conexión a Internet.	2009-03-24	2522	Bueno	24	
Integración Selenium / Maven 2 / Surefire / Cargo / Tomcat 6	Con este tutorial se pretende integrar en nuestro proyecto : Maven, Selenium, Surefire, Cargo y Tomcat 6 con el objetivo de incluir y ejecutar las pruebas de integración dentro del ciclo de vida de Maven.	2009-02-26	1272	Muy bueno	3	
Introducción teórica a XPath	Con este tutorial se pretende enseñar de forma teórica el uso de expresiones XPath pensando sobre todo en su posible uso con el paquete Selenium u otras aplicaciones.	2009-02-26	1622	Muy bueno	4	
Redimensionar Imagenes en Windows Vista	Instalación y uso de ImageResizer	2009-02-26	746	Muy bueno	2	

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.