



Ajax en Java EE

Álvaro Sánchez-Mariscal
alvaro.sanchez@salenda.es

¿Quiénes somos?

- Sobre Salenda.
 - Misión: ser un referente en España en tecnología Java.
 - Líneas de negocio:
 - Formación.
 - Desarrollo de aplicaciones.
 - www.salenda.es

¿Quiénes somos?

- Sobre mí.
 - +6 años de experiencia profesional en Java / Java EE.
 - Instructor oficial de BEA Systems.
 - Mérito “BEA Technical Director”
 - Instructor oficial de Sun Microsystems.
 - Ex-presidente de javaHispano.
 - Ex-vocal del GUL.
 - Fundador de Salenda.



TECHNICAL
DIRECTOR



¿Qué es AJAX?

- **A**synchronous **J**avaScript **A**nd **X**ML
(JavaScript y XML asíncronos),
 - Técnica de desarrollo web para crear aplicaciones interactivas.
- Se ejecuta en el cliente
- Mantiene comunicación asíncrona con el servidor en segundo plano
 - Se realizan cambios en la misma página sin recargarla (más rapidez)

- Desarrollo de Microsoft de Remote Scripting (hace 10 años)
- IFrame (Microsoft Internet Explorer 3, 1996)
 - Carga asíncrona de páginas
- Layers (Netscape 4, 1997)
- Microsoft's Remote Scripting (o MSRS, 1998)
 - Empleo de Applet Java
 - Internet Explorer 4 y Netscape 4
 - Outlook Web Access
- Definición del término AJAX (Febrero 2005)
 - Jesse James Garrett: "A New Approach to Web Applications"

- AJAX es una combinación de tres tecnologías ya existentes:
 - XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño
 - Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario para mostrar e interactuar dinámicamente con la información presentada.
 - El objeto XMLHttpRequest para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto IFrame en lugar del XMLHttpRequest para realizar dichos intercambios.
- XML es el formato usado comunmente para la transferencia de vuelta al servidor
 - Cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano, JSON y hasta EBML.

Web 2.0

- Actualmente la mayoría de las actividades del usuario se desarrollan en la web (Correo, Chats, compartir información...)

Web 1.0	Web 1.5	Web 2.0
Páginas personales	Wikis	Blogs
Email/News	Foros	RSS/Sindicación
MP3	Napster	iTunes
Terraserver	MapQuest	Google Maps
Enciclopedia Británica		Wikipedia
DoubleClick		Google AdSense
Ofoto		Flickr

- Se está produciendo una evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas al usuario final, también conocidas
- RIA (Rich Internet Application):
 - Interfaces intuitivos
 - Sistemas sencillos e intuitivos
 - Comunicación de cambios al usuario

- Tecnologías o desarrollos en las que se sustentan las RIA:
 - Applet
 - Macromedia Flash
 - Java Web Start
 - DHTML
 - DHTML con Iframe
 - AJAX

- API que se encuentra implementado en el navegador y que proporciona los métodos y propiedades necesarios para la comunicación con el servidor mediante HTTP
- Originalmente desarrollado por Microsoft como un objeto ActiveX, disponible desde Internet Explorer 5
- Utilizada por JavaScript, Jscript, VBScript u otros lenguajes de scripting de navegadores web
- Emplea un canal de conexión independiente

Características de XMLHttpRequest

- Comunicación GET/POST
- Documentos pueden ser texto plano/xml
- Trabaja en background
- Número limitado de peticiones
- Permite especificar un manejador para el control de cambios de estado
- Manejador notifica el estado de la petición:
 - Inicializada
 - Iniciada
 - En proceso de retornar la información
 - Operación completada

- Documento XML
- Texto
 - Procesado en el cliente
 - Mostrado directamente
- JavaScript
 - Evaluado en JavaScript mediante 'eval()'
 - JSON, "JavaScript Object Notation":
 - Es un formato ligero para el intercambio de datos
 - Es un subconjunto de la notación literal de objetos de Javascript pero no requiere el uso de Javascript

Ejemplo JSON

Definición de barra de menús usando JSON y XML:

```
{ "menu": {  
  "id": "file",  
  "value": "File",  
  "popup": {  
    "menuitem": [  
      { "value": "New", "onclick": "CreateNewDoc()" },  
      { "value": "Open", "onclick": "OpenDoc()" },  
      { "value": "Close", "onclick": "CloseDoc()" }  
    ]  
  }  
}}
```

```
<menu id="file" value="File">  
  <popup>  
    <menuitem value="New" onclick="CreateNewDoc()" />  
    <menuitem value="Open" onclick="OpenDoc()" />  
    <menuitem value="Close" onclick="CloseDoc()" />  
  </popup>  
</menu>
```

Métodos de XMLHttpRequest

- `open("method", "URL", syn/asyn)`
 - Asigna la URL de destino, el método y otros parámetros opcionales de una petición pendiente
- `send(content)`
 - Envía la petición, opcionalmente se puede enviar una cadena de texto o un objeto DOM
- `abort()`
 - Detiene la petición actual
- `getAllResponseHeaders()`
 - Devuelve todas las cabeceras de la respuesta como pares de etiqueta y valores en una cadena
- `getResponseHeader("headerLabel")`
 - Devuelve el valor de una cabecera determinada
- `setRequestHeader("label", "value")`
 - Asigna un valor al par label/value para la cabecera enviada.

Propiedades de XMLHttpRequest

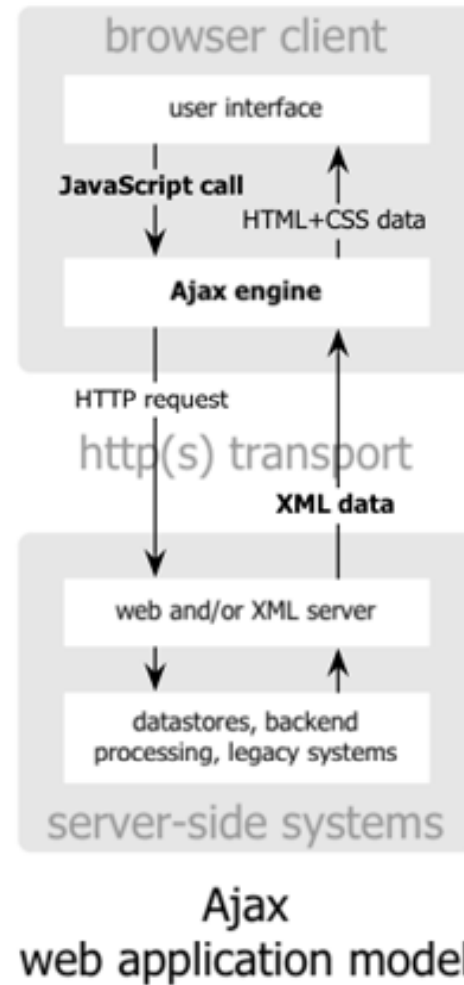
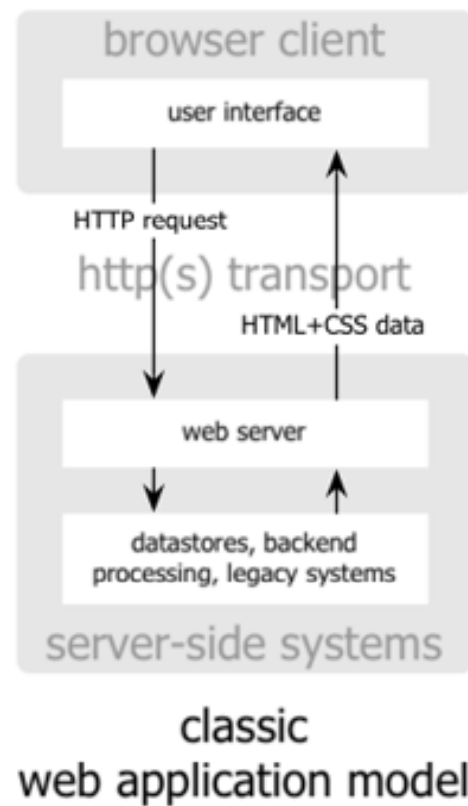
- **onreadystatechange**
 - El manejador del evento llamado en cada cambio de estado del objeto
- **readyState**
 - Indica el estado del objeto o la petición
 - 0 = sin inicializar
 - 1 =cargando
 - 2 = fin de la carga
 - 3 = actualizando la información recibida
 - 4 = Operación completada
- **status**
 - Estado HTTP devuelto por el servidor
 - 404 si la página no se encuentra
 - 200 si todo ha ido bien
- **responseText:**
 - Cadena de texto con los datos devueltos por el servidor
- **responseXML**
 - Objeto DOM devuelto por el servidor
- **statusText**
 - Respuesta del servidor asociada al status (mensaje de texto)

- Alternativas:
 - IFrame oculto
 - Netscape's LiveConnect
 - Microsoft's ActiveX
 - Microsoft's XML Data Islands
 - Macromedia Flash Player
 - Java Applets

Funcionamiento de AJAX

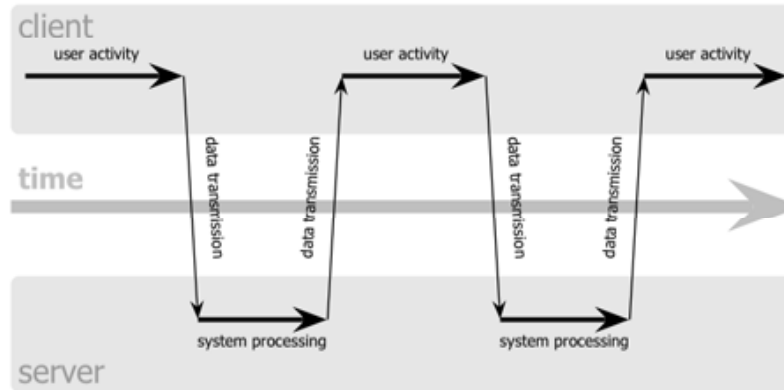
1. Usuario provoca un evento
2. Se crea y configura un objeto XMLHttpRequest
3. El objeto XMLHttpRequest realiza una llamada al servidor
4. La petición se procesa en el servidor
5. El servidor retorna un documento XML que contienen el resultado
6. El objeto XMLHttpRequest llama a la función callback() y procesa el resultado
7. Se actualiza el DOM de la página asociado con la petición con el resultado devuelto

Web tradicional vs AJAX (I)

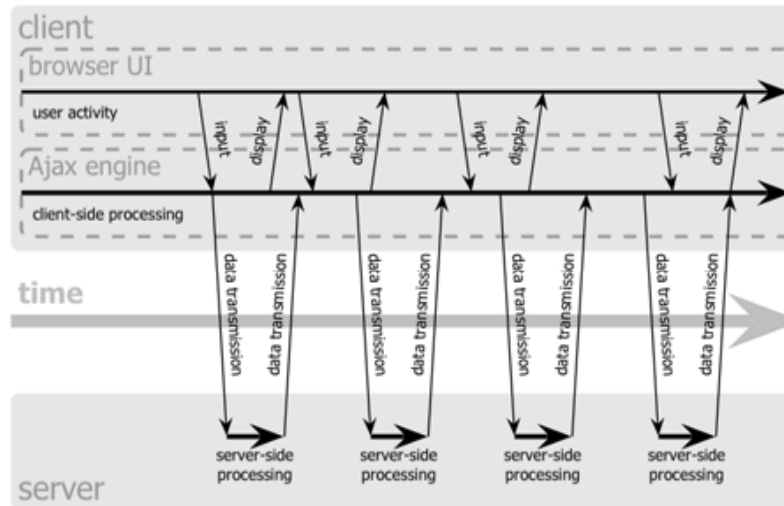


Web tradicional vs AJAX (II)

classic web application model (synchronous)



Ajax web application model (asynchronous)



Navegadores que soportan AJAX

- Microsoft Internet Explorer $\geq$ 5.0
- Navegadores basados en Gecko (versión 7.1)
 - Mozilla, Mozilla Firefox, SeaMonkey, Camino, Flock, Epiphany, Galeon y Netscape
- Navegadores con API KHTML (versión 3.2)
 - Konqueror 3.2, Safari 1.2
- Opera 8.0
- Opera Mobile Browser versión 8.0 (móvil)
- Navegador para S60 de Nokia (móvil)
 - Basado en Safari y API KHTML

- Mayor interactividad
 - Recuperación asíncrona de datos, reduciendo el tiempo de espera del usuario
- Facilidad de manejo del usuario
 - El usuario tiene un mayor conocimiento de las aplicaciones de escritorio
- Se reduce el tamaño de la información intercambiada
- Portabilidad entre plataformas
 - No requieren instalación de plugins, applets de Java, ni ningún otro elemento
- Código público

- Usabilidad: Comportamiento del usuario ante la navegación
 - *Botón de volver atrás del navegador*
 - - Empleo de iframe ocultos para almacenar el historial)
 - - Empleo de fragmento identificador del URL ('#') y recuperación mediante JavaScript
 - *Problema al agregar marcadores/favoritos en un momento determinado de la aplicación*
 - - Empleo del fragmento identificador del URL ('#') y recuperación mediante JavaScript
 - *Problemas al imprimir páginas renderizadas dinámicamente*
- Tiempos de respuesta entre la petición del usuario y la respuesta del servidor
 - Empleo de feedback visual para indicar el estado de la petición al usuario

- JavaScript
 - Requiere que los usuarios tengan el JavaScript activado en el navegador
 - En el caso de Internet Explorer 6 y anteriores, que necesita tener activado el ActiveX (En Internet Explorer 7, se implementa como JavaScript nativo)
 - Como en DHTML, debe comprobarse la compatibilidad entre navegadores y plataformas

- Los desarrolladores necesitan proporcionar opciones para usuarios en otras plataformas o navegadores que no utilicen AJAX

- Validación de datos de formularios en tiempo real
 - Identificadores de usuario, nº de serie, códigos postales u otro código especial que necesite validación en el lado del servidor antes de ser enviado el formulario.
- Autocompletado
 - Direcciones de correo, nombres, ciudades
- Operaciones de detalle
 - Obtener información más detallada de un producto
- GUI avanzadas
 - Controles en árbol, menús, barras de progreso
- Refresco de datos
- Notificaciones del servidor

- Actualizar o eliminar registros
- Expandir formularios web
- Devolver peticiones simples de búsqueda
- Editar árboles de categorías

Frameworks y librerías: Diferencias

- **Librería (o biblioteca)**
 - Proporciona utilidades o extensiones al lenguaje. Hay dos tipos:
 - Javascript
 - Facilitan únicamente el uso de AJAX. Se trata de una biblioteca de javascript que facilita el intercambio de información con el servidor
 - Específicas del lenguaje
 - Genera directamente el JavaScript desde el lenguaje del servidor, de este modo se definen los métodos y se realizan los intercambios de información de forma completamente transparente.
- **Framework**
 - Es muy importante la relación entre los distintos componentes, y en general se requiere adaptar la metodología de trabajo a sus convenciones.

Frameworks o librerías AJAX Salenda

- **Prototype (<http://prototype.conio.net>)**
 - Librería Javascript que agrega utilidades al lenguaje para escribir aplicaciones más complejas con menos código.
 - Aprovecha la maleabilidad de Javascript para enriquecer la funcionalidad de sus objetos base (Object, Array, Number, String)
 - Incluye nuevas clases utilitarias para trabajar con Ajax y un sencillo sistema de herencia para acercar a Javascript a las nociones OOP familiares en Java, Ruby u otros lenguajes orientados a objetos.
 - Prototype no es en si una aplicación, sino una base para escribir aplicaciones más sólidas.
 - De hecho, Prototype es usado para desarrollar librerías especializadas como [Scriptaculous](#) (para Ajax avanzado y efectos cinemáticos) o pequeños Widgets como [LightBox 2](#).

- DOJO Toolkit (<http://dojotoolkit.org/>)
 - Biblioteca JavaScript de código abierto
 - Proporciona un API para el control y manipulación de historial
 - Proporciona en el lado del cliente para la manipulación de URL y marcadores/favoritos
 - Widgets:
 - Ordenar tablas
 - Validación de formularios
 - Menus y barras de menús
 - Google y Yahoo! Maps

Frameworks o librerías AJAX

- XAJAX (<http://www.xajaxproject.org/>)
 - Es una biblioteca PHP de código abierto
 - Abundante documentación
 - Fácil de utilizar:
 - No requiere gran conocimiento de JavaScript
 - Sencillo de emplear:
 - Incluir biblioteca en página PHP
 - Instanciar el objeto 'xajax'
 - Implementación de nuevas funciones en PHP
- XOAD (<http://www.xoad.org/>, antes NAJAX)
 - Biblioteca orientada a objetos basada en PHP
 - Documentación de las clases y tutoriales sencillos
 - Emplea JSON y objetos PHP para la comunicación
 - Soporta eventos del lado del cliente y del servidor

Ejemplos de aplicaciones AJAX (I)

- **Gmail:**
 - Gmail <http://gmail.com>
 - Google Maps <http://maps.google.com/>
 - Google Suggest <http://www.google.com/webhp?complete=1&hl=en>
 - Google Calendar <http://www.google.com/calendar/>
- **Microsoft:**
 - Windows Live Mail <http://mail.live.com>
 - Windows Local Live <http://local.live.com>
 - Amazon Maps <http://maps.a9.com/>
 - HousingMaps <http://www.housingmaps.com/>
 - Nuevo Yahoo! Mail

- Aplicaciones de Escritorio
 - Suite de oficina
 - gOffice (<http://www.goffice.com/>)
 - Thinkfree (<http://online.thinkfree.com/>)
 - Zimbra (<http://www.zimbra.com>)
 - Procesadores de texto
 - Writely (<http://www.writely.com>)
 - FCKeditor (<http://www.fckeditor.net/>)
 - Hoja de cálculo
 - Num Sum (<http://numsum.com>)
 - Numbler (<http://numbler.com/>)
 - Calendarios
 - Kiko (<http://www.kiko.com/>)

– Notas:

- Webnote (<http://www.aypwip.org/webnote/>)

– Lectores RSS:

- Backbase (<http://www.backbase.com/demos/RSS/>)
- Netvibes (<http://www.netvibes.com/>)

– Mensajería instantánea:

- Meebo (<http://www.meebo.com/>)
- IM (<http://ajaxim.unwieldy.net/>)

– Gestor de bases de datos MySQL

- TurboDbAdmin
(<http://www.turboajax.com/turboadbadmin/>)

- AJAXLaunch
 - AJAXWrite (Procesador de textos)
 - <http://www.ajaxlaunch.com/ajaxwrite/>
 - AJAXSketch (Editor gráfico)
 - <http://www.ajaxsketch.com/>
 - Eyespot: Reproductor de Video
 - <http://www.eyespot.com/>
 - ajaxXLS: Hojas de cálculo
 - <http://www.ajaxxls.com/>
 - ajaxTunes: Reproductor de música
 - <http://www.ajaxtunes.com/>

Ejemplo AJAX

- El usuario solicita información de un código postal
 - Al pulsar el botón obtiene el nombre de la localidad

Código postal Solicitar

- Sin recargar la página coloca la respuesta en un DIV.

Código postal Solicitar
La localidad es Getafe

Ejemplo AJAX: Paso 1

- Creación del formulario

```
<form name="f">  
  Código postal  
  <input id="cp" type="text" size="6">  
  <input type="button" value="Solicitar" onclick="getLocalidad()"><br>  
  <span id="respuesta"></span>  
</form>
```

- Captura del evento del cliente

- Click del botón llama a la función javascript del cliente "getLocalidad()"

Ejemplo AJAX: Paso 2

- Creación del objeto XMLHttpRequest
 - Cada navegador instancia el objeto de forma distinta y usamos una función que lo tiene en cuenta

```
function crearXMLReq()
{
    var httpReq=null;
    try { httpReq = new ActiveXObject("Msxml2.XMLHTTP"); } catch(e) { }
    if (!httpReq)
        try { httpReq = new ActiveXObject("Microsoft.XMLHTTP"); } catch(e) { }
    // Si no es IE, será un firefox, Opera, etc.
    if(!httpReq && typeof XMLHttpRequest != "undefined")
        httpReq = new XMLHttpRequest();
    return httpReq;
}
```

Ejemplo AJAX: Paso 3

- Realizar la petición al servidor:
 - Se usa una función anónima para tratar la respuesta, pero se puede invocar a otra función:

```
function getLocalidad()
{
    var httpReq = crearXMLReq();
    var codPostal = document.forms[0].cp.value;
    var url = 'http://misrv.com/servlet/codPostal?cp=' + codPostal;
    httpReq.open("GET", url, true);
    httpReq.onreadystatechange = function () {
        if (httpReq.readyState == 4) {
            var resp = alert(httpReq.responseText);
            document.getElementById('respuesta').innerHTML=resp;
        }
    };
    httpReq.send(null);
}
```

Ejemplo AJAX: Paso 4

- Realizar la parte servidora
 - Puede ser un servlet, un script ASP o PHP, una CGI en C o en Perl, etc.
 - Obtendrá el parámetro “cp” del querystring y devolverá el nombre del pueblo.

- “Ajax: A New Approach to Web Applications”
 - <http://adaptivepath.com/publications/essays/archives/000385.php>
- Wikipedia (<http://wikipedia.org>)
- AJAX Magazine (<http://ajax.phpmagazine.net/>)