

# **Introducción a AJAX**

# ¿Qué es AJAX?

---

Acrónimo de **A**synchronous **J**avaScript **A**nd **X**ML

## Definición

No es ninguna tecnología, ni lenguaje de programación

Es una técnica de desarrollo web que combina varias tecnologías

Consiguiendo una navegación más ágil y rápida, más dinámica.

# Historia

---

- 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"

# Tecnologías empleadas

---

- **XHTML (o HTML) y CSS**  
Presentación de datos
- **Document Object Model (DOM)**  
Mostrar e interactuar dinámicamente con la información
- **XML y XSLT**  
Intercambiar y manipular datos con el servidor web
- **XMLHttpRequest**  
Recuperación y envío de datos de modo asíncrono
- **JavaScript**  
Nexo de unión

# Web 2.0

---

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

## Web 1.0

- Páginas personales
- Email/Grupo de noticias
- Mp3
- Terraserver
- Británica Online
- Ofoto

## Web 1.5

- Wikis
- Foros de discusión
- Napster
- MapQuest

## Web 2.0

- Blogging
- RSS-Sindicación
- iTunes
- Google Maps
- Wikipedia
- Flickr

# Web 2.0

---

- 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

# Web 2.0

---

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

# XMLHttpRequest

---

- 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

# Tipo de Respuesta

---

- 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

# Propiedades de XMLHttpRequest

---

- **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)

# XMLHttpRequest

---

Alternativas:

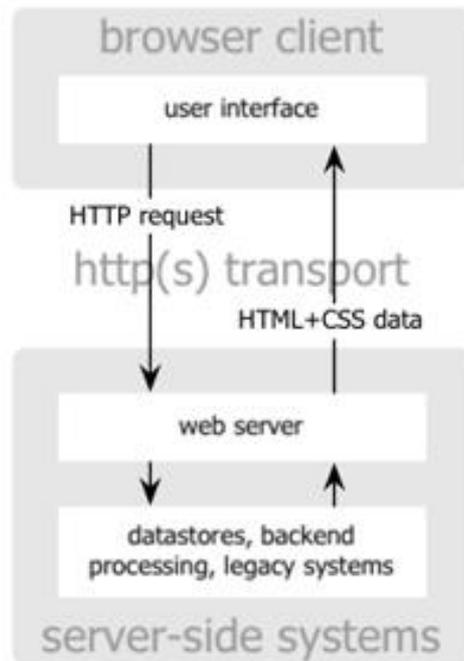
- IFrame oculto
- Netscape's LiveConnect
- Microsoft's ActiveX
- Microsoft's XML Data Islands
- Macromedia Flash Player
- Java Applets

# Funcionamiento de AJAX

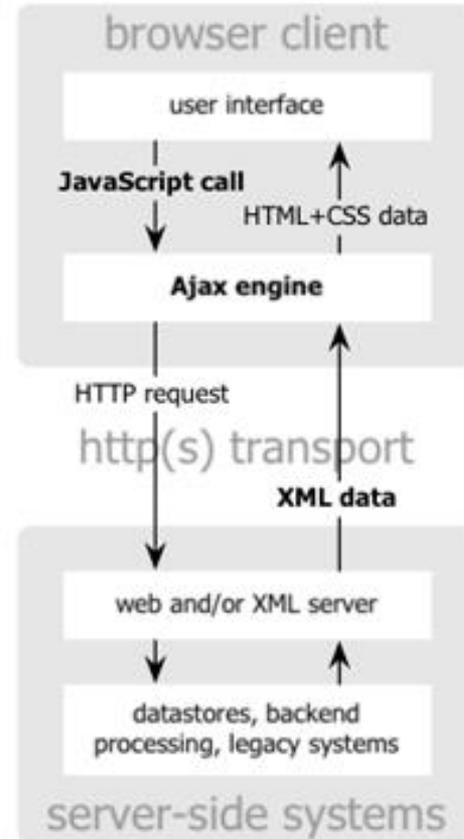
---

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 (1)

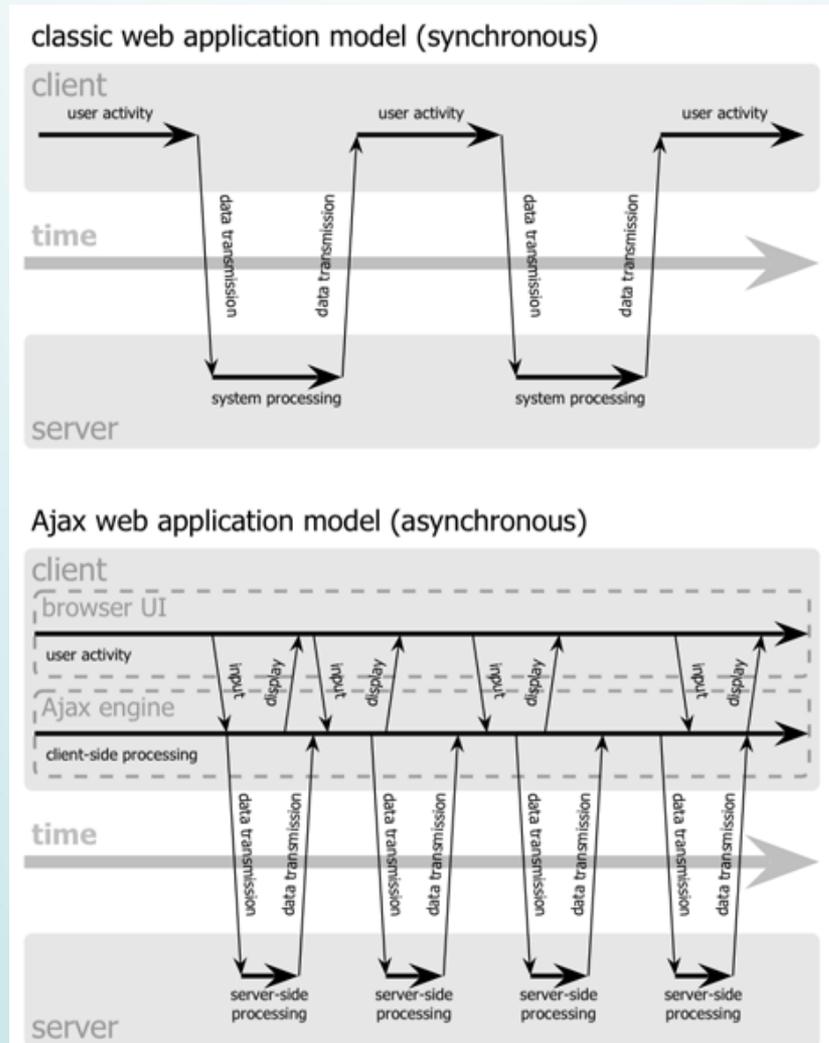


classic  
web application model



Ajax  
web application model

# Web Tradicional vs AJAX (2)



# Navegadores soportados

---

- Microsoft Internet Explorer 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

Navegadores disponibles para de teléfonos móviles:

- Opera Mobile Browser versión 8.0
- Navegador para S60 de Nokia basado en Safari y API KHTML

# Ventajas

---

- 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

# Inconvenientes y Críticas

---

- 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

# Inconvenientes y Críticas

---

- 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

# Accesibilidad

---

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

# USOS DE 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**

# USOS DE AJAX

---

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

# Bibliotecas AJAX

---

- Es posible diferenciar dos tipos de bibliotecas:
  - **Javascript:** Facilitan únicamente el uso de AJAX. Se trata de una biblioteca de javascript que facilita el intercambio de información con el servidor
  - **Específicas de 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.

# Bibliotecas AJAX

---

- 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

# Bibliotecas AJAX

---

- **Prototype (<http://prototype.conio.net/>)**
  - Framework en JavaScript para el desarrollo sencillo y dinámico de aplicaciones web
  - Proyectos basados en Prototype:
    - Ruby on Rails (<http://www.rubyonrails.com/>)
    - [script.aculo.us](http://script.aculo.us), Thomas Fuchs (<http://script.aculo.us/>)
    - Rico (<http://openrico.org/>)
    - Behaviour (<http://www.ripcord.co.nz/behaviour/>)

# Bibliotecas 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 (1)

---

- 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(antes MSN Virtual Earth,  
<http://local.live.com>)
- Amazon Maps (<http://maps.a9.com/>)
- Amazon: (<http://www.a9.com>)
- HousingMaps (<http://www.housingmaps.com/>), con Google Maps
- Nuevo Yahoo! Mail

# Ejemplos de Aplicaciones (2)

---

- 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/>)

# Ejemplos de Aplicaciones (3)

---

- Notas:
  - Webnote(<http://www.aypwip.org/webnote/>)
- Lector 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  
(<http://www.turboajax.com/turbodbadmin/>)

# Ejemplos de Aplicaciones (4)

---

- Sistema Operativo (<http://www.michaelrobertson.com/ajaxos/>)
- 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/>)
  - ajaxOS: Sistema Operativo basado en Linspire (<http://www.myajaxos.com/>)

# ¿Qué es COMET?

---

Técnica de programación Web similar a AJAX

Utiliza XMLHttpRequest para la entrega de datos entre cliente y servidor a través del protocolo HTTP

También conocido como “Server Push o HTTP Push”

## Diferencias con AJAX:

- Emplea una conexión persistente entre el cliente y el servidor web
- La entrega de datos se hace sin que el usuario lo haya solicitado

# COMET

---

## FUNCIONAMIENTO :

1. el cliente no solicita los datos, pero si envía información al servidor
2. el servidor no le responde al cliente con un bloque de datos, se espera a que haya algún evento de lado del servidor para enviar la información

## HISTORIA :

Alex de DojoToolkit.org fue el primero en definir esta técnica, y Dojo es el primer framework que implementa a Comet

# COMET

---

## VENTAJAS :

Comunicación abierta con el servidor → Reduce los tiempos de respuesta

## INCONVENIENTES :

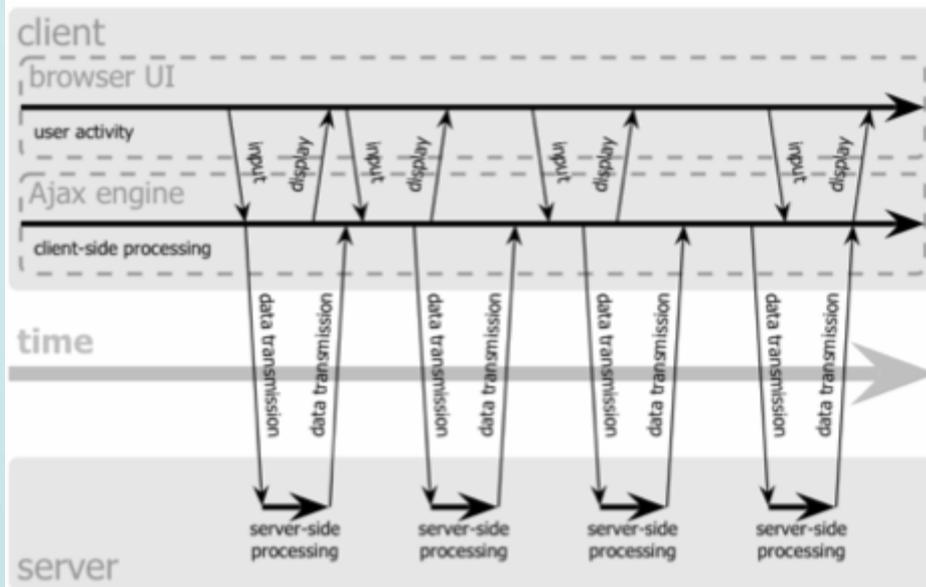
Compartir la misma conexión para diferentes peticiones  
(Conexión no bloqueante)

Capacidad del servidor de mantener el thread asociado a la petición

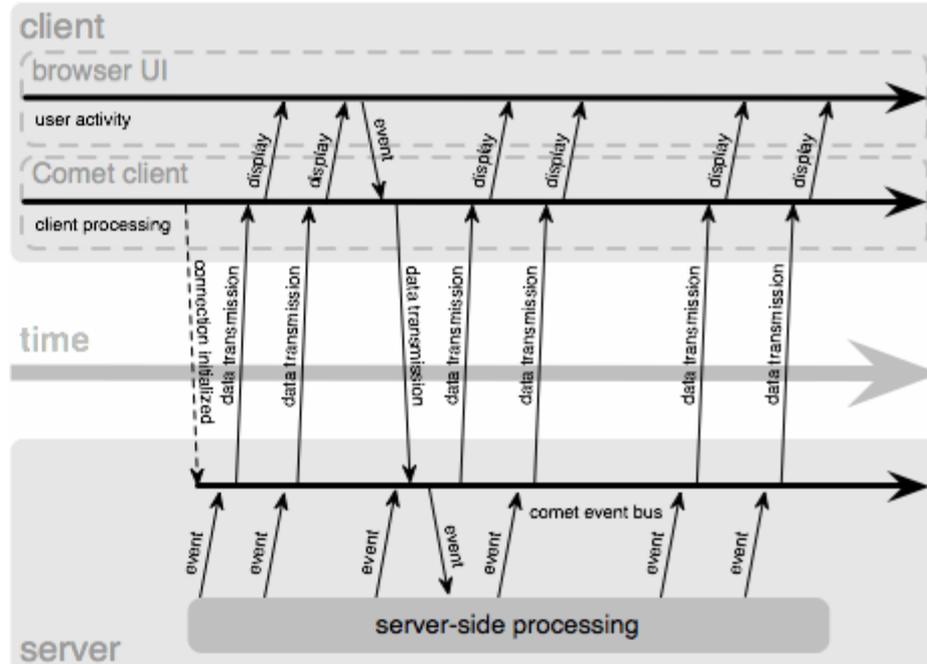
Escalabilidad

# COMET vs AJAX

## Ajax web application model (asynchronous)



## Comet web application model



# EJEMPLOS

---

## MENSAJERÍA INSTANTÁNEA :

- Integración de Gtalk en Gmail
- Jot Live (<http://jotlive.com/>)
- Renkoo (<http://renkoo.com>)
- cgi:irc (<http://cgiirc.sourceforge.net/>)
- Meebo (<http://www.meebo.com/>)