



## Ajax: Un Nuevo acercamiento a las Aplicaciones Web

.: Jesse James Garrett - 2005-06-11

**AJAX parece ser la palabra de moda en el “mundo” del desarrollo de aplicaciones Web, AJAX no es una tecnología, sino la unión de varias tecnologías que juntas pueden lograr cosas realmente impresionantes.**

Hace un tiempo AJAX parece ser la palabra de moda en el “mundo” del desarrollo de aplicaciones Web; de hecho muchos lo escuchan nombrar pero pocos saben que es realmente y, menos aún, saben en donde buscar información clara sobre que es esta nueva “maravilla” de la tecnología que Jesse James Garret publicó en un artículo en Inglés excelente que vale la pena traducir por completo.

¿Porque es tan interesante AJAX? Porque en realidad AJAX no es una tecnología, sino la unión de varias tecnologías que juntas pueden lograr cosas realmente impresionantes como GoogleMaps, Gmail el Outlook Web Access (*ref*) o algunas otras aplicaciones muy conocidas: AJAX, en resúmen, es el acrónimo para **Asynchronous JavaScript + XML** y el concepto es: Cargar y renderizar una página, luego mantenerse en esa página mientras scripts y rutinas van al servidor buscando, en *background*, los datos que son usados para actualizar la página solo re-renderizando la página y mostrando u ocultando porciones de la misma.

La traducción completa a continuación:

### **Ajax: Un Nuevo acercamiento a las Aplicaciones Web**

Por Jesse James Garrett

February 18, 2005

Si algo del actual diseño de interacción puede ser llamado glamoroso, es crear Aplicaciones Web. Después de todo, ¿cuando fue la ultima vez que escuchaste a alguien hablar de diseño de interacción de un producto que no esté en la Web? (Okay, dejando de lado el iPod). Todos los nuevos proyectos cool e innovadores están online.

Dejando de lado esto, los diseñadores de interacción Web no pueden evitar sentirse envidiosos de nuestros colegas que crean software de escritorio. Las aplicaciones de escritorio tienen una riqueza y respuesta que parecía fuera del alcance en Internet. La misma simplicidad que ha permitido la rápida proliferación de la Web también crea una brecha entre las experiencias que podemos proveer y las experiencias que los usuarios pueden lograr de las aplicaciones de escritorio.

Esa brecha se está cerrando. Échenle una mirada a las Google Suggest. Mira la forma en que los términos sugeridos se van actualizando a medida que uno tipea casi instantáneamente. Ahora mire Google Maps. Hace zoom. Usen el cursor para agarrar el mapa y navegarlo un poco. Otra vez, todo sucede casi instantáneamente, sin esperar que las paginas se recarguen.

Google Suggest y Google Maps son dos ejemplos de un nuevo acercamiento a las aplicaciones Web, que nosotros en Adaptive Path hemos denominado AJAX. El nombre es una abreviación o acrónimo para Asynchronous JavaScript + XML, y ello representa un cambio fundamental en que es posible en la Web.

## Definiendo Ajax

Ajax no es una tecnología. Es realmente muchas tecnologías, cada una floreciendo por su propio mérito, uniéndose en poderosas nuevas formas. AJAX incorpora:

- presentación basada en estándares usando XHTML y CSS;
- exhibición e interacción dinámicas usando el Document Object Model ;
- Intercambio y manipulación de datos usando XML and XSLT ;
- Recuperación de datos asincrónica usando XMLHttpRequest ;
- y JavaScript poniendo todo junto.

El modelo clásico de aplicaciones Web funciona de esta forma: La mayoría de las acciones del usuario en la interfaz disparan un requerimiento HTTP al servidor web. El servidor efectúa un proceso (recopila información, procesa números, hablando con varios sistemas propietarios), y le devuelve una pagina HTML al cliente. Este es un modelo adaptado del uso original de la Web como un medio hipertextual, pero como fans de "The Elements of User Experience" sabemos, lo que hace a la Web buena para el hipertexto, no la hace necesariamente buena para las aplicaciones de software.

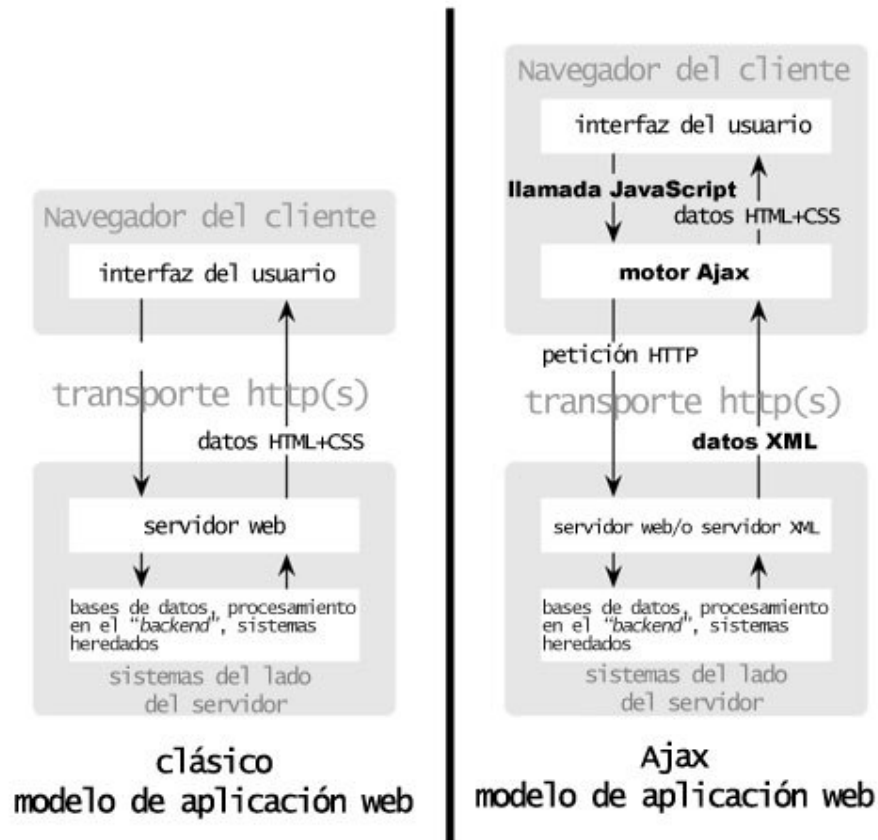


Figura 1: El modelo tradicional para las aplicaciones Web (izq.) comparado con el modelo

*de AJAX (der.).*

Este acercamiento tiene mucho sentido a nivel técnico, pero no lo tiene para una gran experiencia de usuario. Mientras el servidor esta haciendo lo suyo, que esta haciendo el usuario? Exacto, esperando. Y, en cada paso de la tarea, el usuario espera por mas.

Obviamente, si estuviéramos diseñando la Web desde cero para aplicaciones, no querríamos hacer esperar a los usuarios. Una vez que la interfaz esta cargada, porque la interacción del usuario debería detenerse cada vez que la aplicación necesita algo del servidor? De hecho, porque debería el usuario ver la aplicación yendo al servidor?

### **Como es diferente AJAX**

Una aplicación AJAX elimina la naturaleza “arrancar-frenar- arrancar-frenar” de la interacción en la Web introduciendo un intermediario -un motor AJAX- entre el usuario y el servidor. Parecería que sumar una capa a la aplicación la haría menos reactiva, pero la verdad es lo contrario.

En vez de cargar un pagina Web, al inicio de la sesión, el navegador carga al motor AJAX (escrito en JavaScript y usualmente “sacado” en un frame oculto). Este motor es el responsable por renderizar la interfaz que el usuario ve y por comunicarse con el servidor en nombre del usuario. El motor AJAX permite que la interacción del usuario con la aplicación suceda asincrónicamente (independientemente de la comunicación con el servidor). Así el usuario nunca estará mirando una ventana en blanco del navegador y un icono de reloj de arena esperando a que el servidor haga algo.

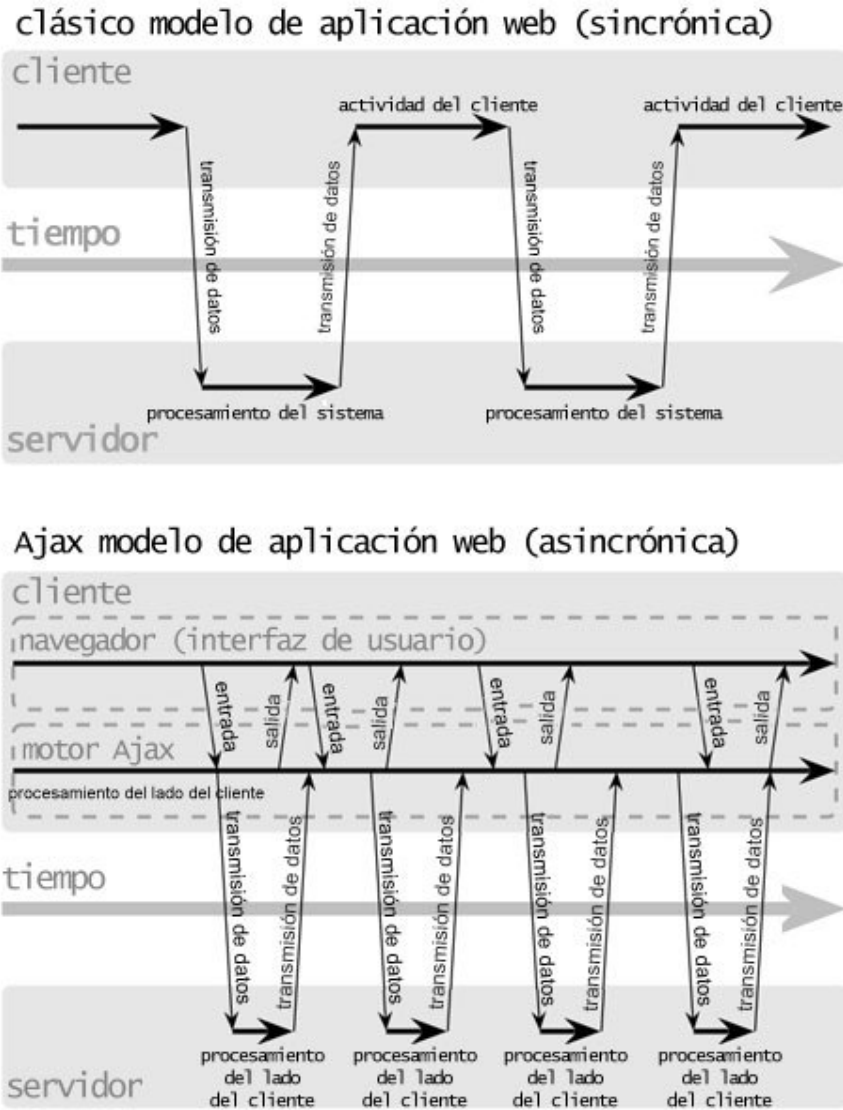


Figura 2: El patrón de interacción sincrónica de una aplicación Web tradicional (arriba) comparada con el patrón asincrónico de una aplicación AJAX (abajo).

Cada acción de un usuario que normalmente generaría un requerimiento HTTP toma la forma de un llamado JavaScript al motor AJAX en vez de ese requerimiento. Cualquier respuesta a una acción del usuario que no requiera una vuelta al servidor (como una simple validación de datos, edición de datos en memoria, incluso algo de navegación) es manejado por su cuenta. Si el motor necesita algo del servidor para responder (sea enviando datos para procesar, cargar código adicional, o recuperando nuevos datos) hace esos pedidos asincrónicamente, usualmente usando XML, sin frenar la interacción del usuario con la aplicación.

### Quien está usando Ajax

Google está haciendo una significativa inversión en el acercamiento Ajax. Todos los grandes productos que Google ha introducido en el último año ([Orkut](#), [Gmail](#), la última versión de [Google Groups](#), [Google Suggest](#), y [Google Maps](#)) son aplicaciones Ajax. (Para datos más técnicos de estas implementaciones Ajax, lean estos excelentes análisis de

Gmail, Google Suggest, y Google Maps.) Otros están siguiendo la tendencia: muchas de las funciones que la gente ama en Flickr dependen de Ajax, y el motor de búsqueda de Amazon A9.com aplica tecnologías similares.

Estos proyectos demuestran que Ajax no es solo técnicamente importante, sino también prácticos para aplicaciones en el mundo real. Esta no es otra tecnología que solo trabaja en un laboratorio. Y las aplicaciones Ajax pueden ser de cualquier tamaño, de lo más simple, funciones simples como Google Suggest a las muy complejas y sofisticadas como Google Maps.

En Adaptive Path, estuvimos haciendo nuestro propio trabajo con Ajax en los últimos meses, y estamos descubriendo que solo raspamos la superficie de la rica interacción y respuesta que que las aplicaciones Ajax puede proveer. Ajax es un desarrollo importante para las aplicaciones Web, y su importancia solo va a crecer. Y como hay tantos desarrolladores que ya conocen como usar estas tecnologías, esperamos ver mas empresas y organizaciones siguiendo el liderazgo de Google en explotar la ventaja competitiva que Ajax provee.

### **Mirando adelante**

Los mayores desafíos al crear aplicaciones Ajax no son técnicas. Las tecnologías centrales son maduras, estables y bien conocidas. En cambio, los desafíos son para los diseñadores de estas aplicaciones: olvidar lo que creemos saber sobre las limitaciones de la Web, y comenzar a imaginar un rango más amplio y rico de posibilidades. Va a ser divertido.

*Nota: Artículo original escrito por Jesse James Garrett, uno de los fundadores de Adaptive Path y traducido por Mariano Amartino (Denken Uber)*

**Jesse James Garrett**

Director de Estrategia para la experiencia del usuario y fundador de Adaptive Path.

<http://www.adaptivepath.com/team/jjg.php>

**Opina y visita el artículo online:**

<http://www.maestrosdelweb.com/editorial/ajax/>

Articulo Publicado y traducido Originalmente en

