

## Ajax: Uma Nova Técnica para Aplicações Web

Se alguma das atuais interações no design podem ser chamadas de “glamurosas”, é a criação de aplicações Web. Apesar de tudo, quando foi à última vez que você ouviu alguém se entusiasmar com a interação de design de um produto que não está na Web? (alem do [iPod](#)). Todos os novos projetos inovadores legais são on-line.

A respeito disso, desenvolvedores de interatividade na web não podem ajudar, mas sentem uma pontada de inveja de nossos colegas que criam softwares de desktop. Aplicações desktop têm uma riqueza e comunicabilidade que tem parecido fora de alcance na Web. A mesma simplicidade que permite a proliferação rápida da Web também cria uma barreira entre as experiências que podemos prover e as experiências que os usuários podem conseguir de uma aplicação desktop.

Esta barreira está desaparecendo. Dê uma olhada no [Google Suggest](#). Observe o modo como atualizam os termos sugeridos conforme você digita (digite vagarosamente para ver as mudanças). Agora olhe para o [Google Maps](#). Aproxime. Use seu cursor para capturar o mouse e rolá-lo um pouco. De novo, tudo acontece quase instantaneamente, sem esperar pelo carregamento de páginas.

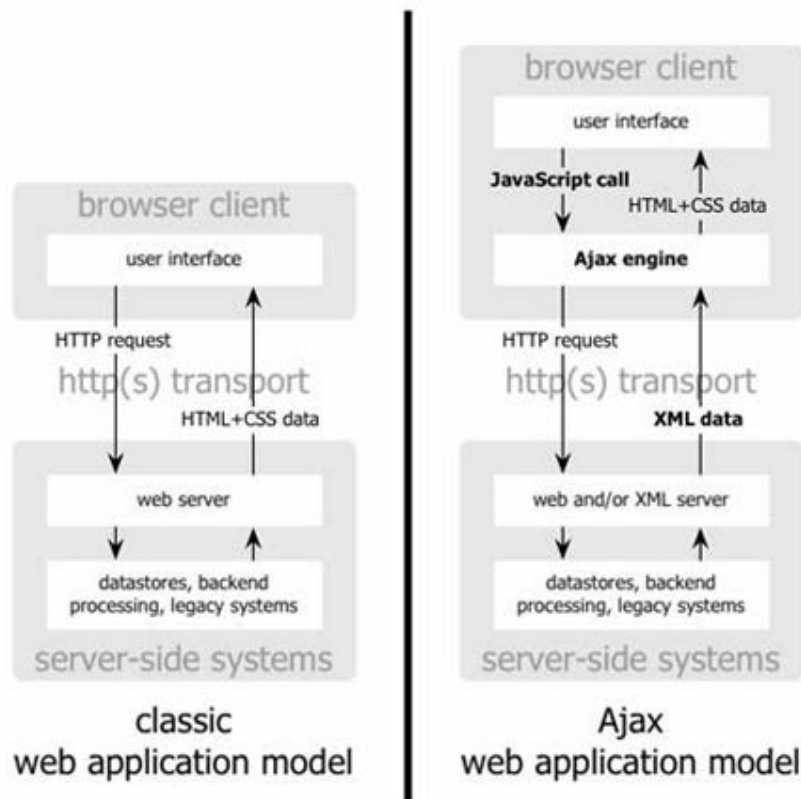
[Google Suggest](#) e [Google Maps](#) são dois exemplos de uma nova técnica para aplicações web que a [Adaptive Path](#) tem chamado de [Ajax](#). O nome é a abreviação para **Asynchronous Javascript and XML**, e representa um passo fundamental para o que é possível na Web.

### Definindo Ajax

**Ajax não é uma tecnologia** (vamos prestar atenção nisso). É, na verdade são várias tecnologias, cada uma atuando da sua própria maneira, tornando-se juntas uma poderosa possibilidade. Ajax incorpora:

1. [Apresentação baseada nas Web Standards usando XHTML e CSS](#).
2. Exibição e interação dinâmicas usando [Document Object Model \(DOM\)](#);
3. Troca e manipulação de dados usando [XML e XSLT](#);
4. Retorno de dados assincronamente usando [XMLHttpRequest](#);
5. E [JavaScript](#) interligando tudo.

O modelo clássico de uma aplicação web trabalha como essa: A maior parte das ações do usuário na interface faz uma requisição HTTP de volta ao servidor web. O servidor processa retornando dados, interpretando números, conversando com vários sistemas legados, e então retorna uma página (X)HTML para o cliente. Esse é um modelo adaptado do uso original da Web como um hipertexto médio, mas como os fãs do [“The Elements of User Experience”](#) sabem, o que faz a web boa para hipertexto não a torna, necessariamente, boa para aplicações de software.



**Figura 1:** O modelo tradicional para aplicações web(esquerda) comparado ao modelo Ajax (direita).

Essa técnica faz muito sentido, mas não para uma grande experiência do usuário. Enquanto o servidor está fazendo esta processando as requisições, o que o usuário está fazendo? Está certo, **esperando**. E a cada etapa da tarefa o usuário espera mais um pouco.

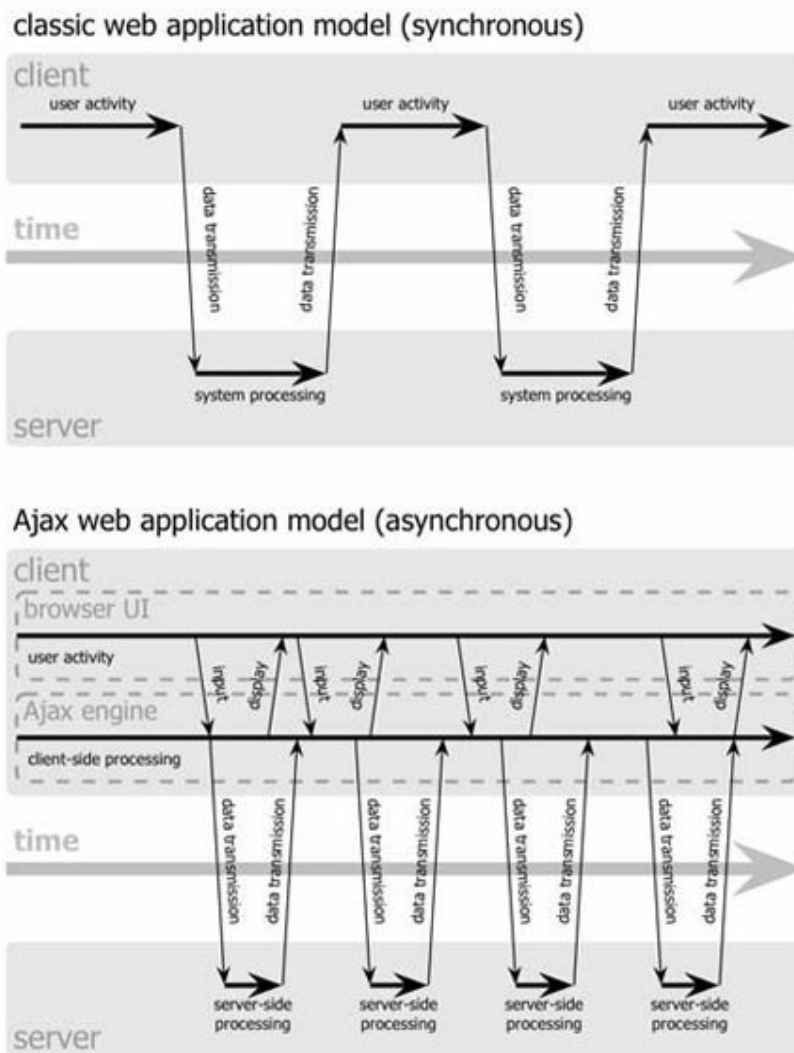
Obviamente, se desenvolvemos a Web para trabalhar com aplicações, nós não faremos o usuário esperar. Uma vez que uma interface é carregada, por que a interação do usuário deve sofrer uma parada a cada vez que a aplicação precisa de algo do servidor? De fato, por que o usuário deve ver a aplicação ir para o servidor afinal?

### Como o Ajax é diferente

Uma aplicação Ajax elimina o natural carrega-para carrega-para da atual interação na Web, introduzindo um intermediário, ou seja, uma ferramenta Ajax entre o usuário e o servidor. Parece que se adicionarmos uma camada para a aplicação a tornaria menos comunicativa, mas o oposto disso é verdade.

Em vez de carregar uma página web, no início da sessão, o browser carrega uma ferramenta Ajax, escrita em JavaScript e geralmente colocada sobre um frame escondido. Esta ferramenta é responsável por renderizar a interface que o usuário vê e pela comunicação com o servidor em prol do usuário. A ferramenta Ajax permite que a interação do usuário com a aplicação aconteça assincronamente, independente da

comunicação com o servidor. Então o usuário nunca encarará uma janela branca do navegador e um ícone de ampulheta, esperando pelo servidor para fazer algo.



**Figura 2:** O padrão de interação síncrono de uma aplicação web tradicional (topo) comparada ao padrão assíncrono de uma aplicação Ajax (embaixo).

Toda ação do usuário que normalmente geraria uma requisição HTTP toma, ao invés disso, a forma de uma chamada JavaScript para a ferramenta Ajax. Qualquer resposta para a ação do usuário que não requer uma viagem de volta ao servidor, como validações simples de dados, manipulação de dados na memória, e até mesmo alguma navegação é tratada pela própria ferramenta. Se a ferramenta precisa de algo do servidor para responder, ou seja, dados submetidos para processamento, carregamento de código de interfaces adicionais, ou retorno de novos dados, a ferramenta faz essas requisições assincronamente, geralmente usando XML, sem parar a interação do usuário com a aplicação.

## Quem está usando Ajax

Google está fazendo um gigantesco investimento no desenvolvimento de aplicações Ajax. Todos dos maiores produtos do Google introduzidos nos últimos anos, como [Orkut](#)(?), [Gmail](#), as últimas versões betas do [Google Groups](#), [Google Suggest](#) e [Google Maps](#), são aplicações Ajax. (Para saber mais sobre essas técnicas doidas e descrições dessas implementações Ajax, cheque estas excelentes análises do [Gmail](#), [Google Suggest](#) e [Google Maps](#)). Outros estão seguindo essas características: algumas das funcionalidades que as pessoas amam no [Flickr](#) dependem do Ajax, e a ferramenta de busca [A9](#) da Amazon aplica técnicas similares.

Estes projetos demonstram que Ajax não é só tecnicamente possível, mas também implementado por aplicações no mundo real. Esta não é outra tecnologia que trabalha somente em laboratório. Aplicações Ajax podem ser de qualquer proporção, desde algo muito simples, como as funções de escolha do Google Suggest até algo muito complexo e sofisticado como Google Maps.

Na Adaptive Path, nós temos feito nosso próprio trabalho com Ajax através dos últimos meses, e apesar de tudo que já fizemos, somente arranhamos a superfície das interações ricas e comunicabilidade que as aplicações Ajax podem prover. Ajax é uma importante desenvoltura para aplicações Web, e essa importância tende somente a crescer. E devido a existirem tantos desenvolvedores que já sabem como usar essas tecnologias, nós esperamos ver muito mais organizações seguindo a direção do Google, recolhendo as vantagens competitivas que o Ajax oferece.

## **Indo Adiante**

Os enormes desafios em criar aplicações Ajax não são técnicas. O coração das tecnologias Ajax é maduro, estável, e bem conhecido. Ao contrário disso, os desafios são para os desenvolvedores dessas aplicações: esquecer o que nós achamos saber sobre as limitações web, e começar a imaginar uma longa e rica gama de possibilidades.

Isto está se tornando divertido.

---

## **Ajax P&R (Perguntas e Respostas)**

Março 13, 2005: Desde quando o foi publicado a artigo de Jesse, recebemos uma enorme quantidade de correspondência dos leitores com questões sobre Ajax. Neste P&R, Jesse responde algumas dessas questões mais comuns.

A Adaptive Path inventou o Ajax? Foi o Google? Adaptive Path ajudou a construir as aplicações Ajax do Google?

Nem a Adaptive Path, nem o Google inventaram o Ajax. Os produtos recentes do Google são simplesmente os exemplos mais aperfeiçoados de aplicações Ajax. Adaptive Path não esteve envolvida no desenvolvimento das aplicações Ajax do Google, mas nós estamos trabalhando com Ajax para muitos de nossos outros clientes.

A Adaptive Path está vendendo componentes Ajax ou comercializando o nome? Onde eu posso baixá-lo?

R. Ajax não é algo que você possa baixar. É uma técnica - um modo de pensar sobre a arquitetura de aplicações web usando certas tecnologias. Nem o nome Ajax, nem as técnicas são proprietárias da Adaptive Path.

O Ajax é apenas outro nome para XMLHttpRequest?

Não. XMLHttpRequest é somente parte do conjunto Ajax. XMLHttpRequest é o componente técnico que torna possível a comunicação assíncrona com o servidor; Ajax é nosso nome para as técnicas abrangentes descritas no artigo, que incluem não somente o XMLHttpRequest, mas também CSS, DOM, e outras tecnologias.

Por que vocês sentiram a necessidade de dar a isso um nome?

Eu precisei de algo mais curto que “Assíncrono JavaScript+CSS+DOM+XMLHttpRequest” para usar quando discutia essa técnica com clientes.

Técnicas para comunicação assíncrona com o servidor têm estado por perto há anos. O que faz do Ajax uma “nova” técnica?

O que é novo é o uso proeminente destas técnicas nas aplicações no mundo real para mudar o modelo de interação fundamental da Web. Ajax está pegando agora porque estas tecnologias e o entendimento da indústria de como usá-las mais eficientemente levam tempo para serem desenvolvidas.

O Ajax é uma plataforma tecnológica ou é um estilo de arquitetura?

São ambos. Ajax é um conjunto de tecnologias sendo usadas juntas de um modo particular.

Pra que tipos de aplicação o Ajax é melhor adaptado?

Não sabemos ainda. Porque essa é uma técnica relativamente nova, nosso entendimento de onde Ajax pode melhor ser aplicado está ainda na sua infância. Algumas vezes o modelo de aplicação web tradicional é a mais apropriada solução para o problema.

Isto significa que a Adaptive Path é anti-Flash?

Não totalmente. Macromedia é um cliente da Adaptive Path, e temos sido grandes patrocinadores da tecnologia Flash. Com o amadurecimento do Ajax, esperamos que algumas vezes Ajax torne-se a melhor solução para um problema particular, e algumas vezes Flash seja a melhor solução. Nós estamos também interessados em explorar de que modo as tecnologias podem ser misturadas (como no caso do Flickr, que usa ambos).

Ajax tem acessibilidade significativa ou limitações de compatibilidade dos browsers?

Aplicações Ajax interrompem o botão Voltar? Ajax é compatível com REST? Existem considerações de segurança no desenvolvimento Ajax? Aplicações Ajax podem ser feitas para trabalhar para usuários que têm JavaScript desativado?

A resposta para todas essas questões é “talvez”. Muitos desenvolvedores já estão trabalhando sobre essas indagações. Nós achamos que existe mais trabalho a ser feito para estipular todas essas limitações do Ajax, e esperamos que a comunidade de desenvolvimento Ajax publique mais artigos como este.

Alguns dos exemplos do Google que você citou não usam XML em tudo. Eu tenho que usar XML e/ou XSLT em uma aplicação Ajax?

Não. XML é o meio mais completo desenvolvido de conseguir dados dentro e fora de um cliente Ajax, mas não há razões para você não poder obter os mesmos efeitos usando tecnologias como Notação de Objeto JavaScript ou algum meio similar de estruturamento de dados para intercâmbio.

Aplicações Ajax são mais fáceis de desenvolver que aplicações web tradicionais?

Não necessariamente. Aplicações Ajax inevitavelmente envolvem código JavaScript complexo rodando no cliente. Tornar este complexo código eficiente e livre de bugs não é uma tarefa para ser tomada negligentemente, e ferramentas de desenvolvimento melhores e frameworks serão necessários para ajudar-nos neste desafio.

Aplicações Ajax sempre fornecem uma experiência melhor que as aplicações web tradicionais?

Não necessariamente. Ajax fornece interações mais flexíveis. Entretanto, quanto mais poder nós temos, mais cuidado devemos ter em usá-lo. Devemos ser cautelosos ao usar Ajax para melhorar as experiências do usuário nas nossas aplicações, não esquecer disto.