

10 Lugares Onde Você Deveria Usar Ajax

por: Alex Bosworth

traduzido por: Vanessa Schissato

Avaliação (Vote clicando) - 71 voto(s)

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ (6.86)
1 2 3 4 5 6 7 8 9 10

1 de dezembro de 2005

Já está se passando um ano agora desde que o GMail mudou o modo como todos imaginavam as aplicações web.

Agora é oficialmente importuno usar aplicações web que não tenham substituído as maravilhosas funcionalidades html pelo arrebatador Ajax.

Aqui estão ocasiões onde Ajax deveria ser requerido à partir de agora em aplicações web:

- **Interações levadas por formulários.**

Formulários são lentos. Muito lentos. Editando um texto num bookmark do del.icio.us? Clique sobre o link de editar para carregar a página do formulário de edição do bookmark, então edite o campo e então submeta para esperar que a submissão vá, então retorne para a página anterior e role pra baixo para encontrar o bookmark e ver se o texto ficou certo. Ajax? Clique no link de editar para instantaneamente começar a mudar o texto, clique no botão submit para enviar assincronamente as mudanças e rapidamente ver no lugar em que elas mudaram, sem recarregar a página inteira.

- **Navegação em árvore com grande hierarquia**

Primeiro de tudo, aplicações com navegações em árvore com grandes hierarquias são geralmente um pesadelo. Topologias horizontais simples e busca/indexação trabalham bem na maioria das circunstâncias. Mas se a aplicação realmente precisa disso, use Javascript para administrar a topologia ui [N/T: *user interface*], e Ajax para diminuir a carga do servidor devido ao carregamento lento de dados da grande hierarquia. Por exemplo: este é um modo que consome muito tempo pra ler listas de discussões ao serem clicados e carregarem completamente novas páginas para se ver uma única linha de resposta [N/T: *um exemplo é o fórum do phpbrasil, com várias listas, com respostas, hierarquia das respostas,etc...*].

- **Comunicação usuário-a-usuário rápida**

Em uma aplicação de postagem de mensagens que cria discussões imediatas entre pessoas, o que realmente enche é forçar o usuário a atualizar a página de novo e de novo para ver uma resposta. Respostas deveriam ser instantâneas, usuários não deveriam ter a necessidade de atualizar a página obcessivamente. Até mesmo o Gmail, que aperfeiçoa as ultrapassadas atualizações de caixas de e-mail do hotmail/yahoo, o sintoma do refresh da caixa de e-mail, não expande o Ajax o suficiente ainda em termos de notificar novos e-mails instantaneamente.

- **Votação, caixas Sim/Não, envio de avaliações**

É realmente muito ruim não existirem exemplos de UI consistentes para submissão Ajax, porque submeter uma votação ou uma resposta sim/não é muito menos doloroso quando a submissão é tratada através do Ajax. Por reduzir o tempo e o impacto de clicar sobre as coisas, aplicações Ajax tornam-se muito mais interativas - se leva uns 40 segundos pra registrar um voto, a maioria das pessoas provavelmente deixam menos opiniões do que elas gostariam. Se levasse 1 segundo pra votar, uma grande porcentagem das pessoas estaria disposta a votar. (Eu tenho 2008 avaliações de filmes no Netflix contra 210 avaliações no IMDb.com).

- **Filtrando e aperfeiçoando manipulações de dados**

Aplicando um filtro, sorteando por data, sorteando por data e nome, permutando dados e cancelando filtros, etc. Qualquer grande manipulação de dados deveria realmente ser feita em Javascript ao invés de através de uma série de requisições ao servidor. Encontrar e manipular

muitos dados é difícil o bastante sem ter que esperar 30 segundos entre cada visualização das mudanças, Ajax pode realmente acelerar isso.

• Sugestão de textos entrados frequentemente / Autocompletar

Entrar com as mesmas frases de texto ou frases de texto previsíveis é algo do qual o software/javascript pode ser muito bom para salvá-lo. É muito útil no de.icio.us e GMail, para adicionar rapidamente bookmarks/endereços de e-mail.

Para aplicações de uso pesado como um cliente de webmail ou em blogreaders, usuários têm a luxúria de ganhar tempo para aprender novos conceitos UI, e a frustração de interagir com interfaces lentas. Este tipo de aplicação é uma perfeita oportunidade para alavancar Ajax em qualquer lugar. Quanto maior a frequência com que usuários usam a aplicação, mais Ajax deveria estar turbinando este uso.

De qualquer forma, para a maioria das aplicações web, isto não torna sensato usar Javascript para tudo ou qualquer coisa. Ajax nitidamente só ajuda realmente em um grupo limitado de circunstâncias; a web já trabalhava bem o suficiente antes do Ajax e existem muitas [armadilhas](#) e empecilhos para usar Ajax no desenvolvimento web. Um simples weblog html funciona muito bem sem estar sendo gerado dinamicamente no cliente com uma corrente de mensagens assíncronas do servidor. Simples HTML antigos também funcionam muito bem para documentos, ou navegação entre documentos. Aplicações simples ou raramente usadas podem seguir bem sem enfiar um cacho de Javascript nelas.

Aqui estão alguns lugares onde Ajax não deveria ser usado:

• Formulários simples

Apesar dos formulários serem simplesmente os grandes beneficiados da "Ajaxiação", um simples formulário de comentários, ou formulário de envio de regras, ou outro formulário raramente usado não se beneficia das submissões dirigidas do Ajax. Geralmente, se um formulário não é muito usado, ou se é necessário trabalhar de forma simples, Ajax não é útil.

• Busca

LiveSearch em blogs é mais aborrecedor que útil. Existe uma razão para o [Google Suggest](#) permanecer na versão beta e não partir para a página principal do Google. Pesquisar no [Start.com](#), [Live.com](#) não possibilita usar o botão Voltar para ver buscas anteriores, ou páginas posteriores. É provável que ninguém tenha conseguido isto direito ainda, mas conseguir isto é trabalhoso o bastante para que geralmente não seja uma boa idéia, ou mais problemático que compensador.

• Navegação básica

Em geral, administrar um site/aplicação de navegação básica usando Ajax é uma péssima idéia. Por que alguém iria querer gastar tempo escrevendo códigos para simular o comportamento do browser quando ele poderia gastar tempo tornando suas aplicações melhores? Para navegação básica entre documentos, Javascript não ajuda.

• Substituindo uma grande quantidade de texto

Ajax poupa uma atualização completa da página, então, pequenos pedaços da página podem ser atualizados mais dinamicamente. Mas se praticamente tudo numa página está mudando, por que não requisitar simplesmente uma nova página ao servidor?

• Mostrando manipulação

Mesmo que pareça que Ajax é puramente uma tecnologia UI, não é. É, na verdade, uma tecnologia de sincronização, manipulação e transferência de dados. Para aplicações web mais limpas e estáveis, é uma boa idéia não ter Ajax/Javascript manipulando a interface diretamente pra tudo. Javascript pode permanecer separado e só manipular o DOM XHTML/HTML, com CSS ditando como o UI mostra esses dados. Veja [este post](#) para um exemplo simples do uso de CSS em vez do Javascript para controlar a exibição.

• Widgets inúteis

Sliders [N/T: menus deslizantes. [Veja um exemplo](#)], drag e drops [N/T: arrastar e soltar], bouncies, mouses gestures [N/T: executar comandos com gestos do mouse. [Saiba mais](#)], o que for. A maioria destes widgets poderiam ser substituídos por controles mais intuitivos, ou remover tudo

em favor da simplicidade. Na escolha de uma cor, talvez um widget slider seja útil para a escolha do tom exato. Mas para escolher uma faixa de preço numa loja, um slider para escolha dos centavos seria uma chacina.

OK, eu menti no título, só publiquei 6 lugares onde Ajax deveria ser usado e 6 lugares onde provavelmente não deveria. Mas olhe para este artigo mais [tetradimensionalmente](#); eu já coloquei [uma página wiki no SWik para lugares onde usar Ajax](#), e eu ou qualquer outro pode encontrar mais lugares onde Ajax seja útil com o passar do tempo.