

Problemas com App Cache e conteúdo dinâmico

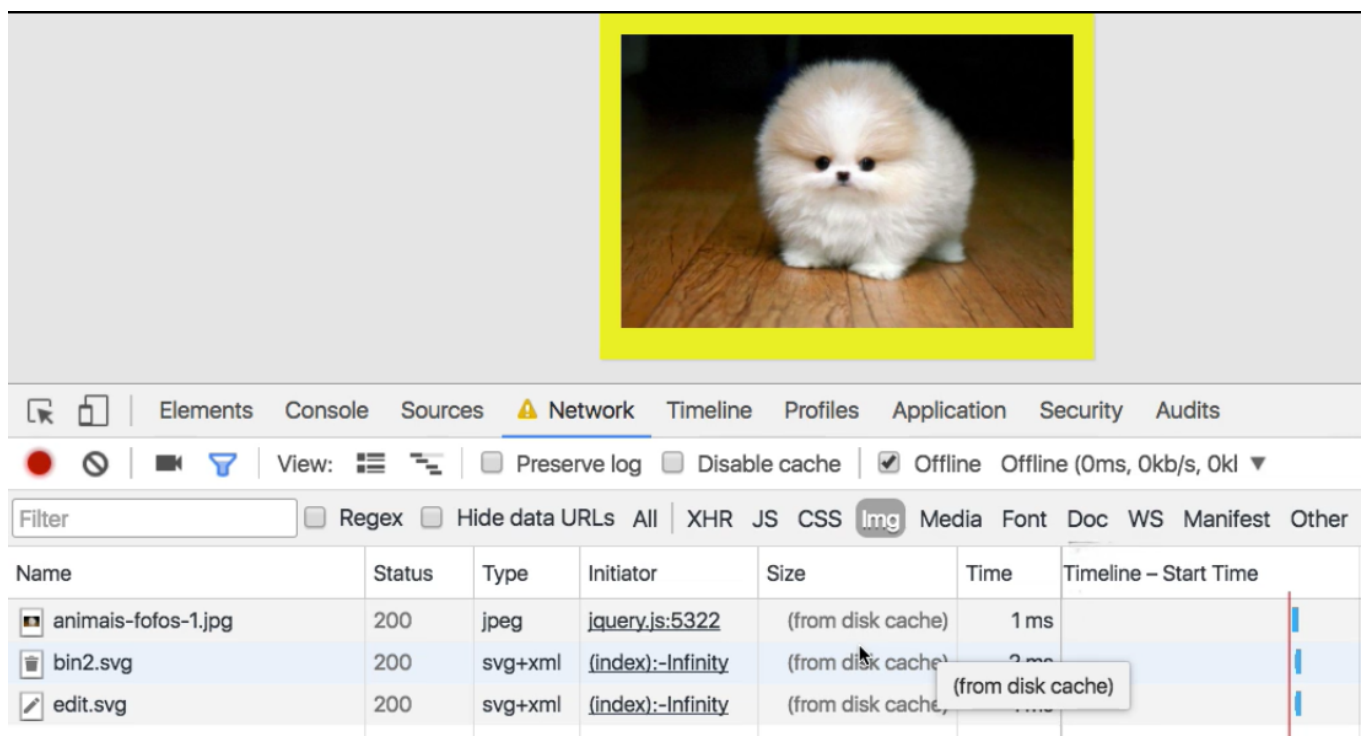
Transcrição

Agora que mostrei para vocês que existe uma nova funcionalidade no nosso mural, que é adicionar imagens, precisamos pensar em como essa imagem funcionará quando estivermos offline, já que atualmente nossos cartões são salvos no `localStorage`. Como isto acontece?

Vou à aba "Application", no `localStorage`, e verei quais cartões estão salvos em meu usuário. Um deles provavelmente estará com esta imagem, não ela em si, e sim sua url, o qual será editado conforme minha vontade ou necessidade (ou seja, não é a imagem em si que será editada).

Isso quer dizer que sempre que o cartão aparecer na tela, a página está pedindo a imagem para o servidor. Quando offline, isto significa que a solicitação não chegará ao servidor. Faço um teste ficando offline e recarregando a página e... Ele carregou a imagem!

Estou na aba "Network", utilizando o filtro "Img" (Imagem), e é possível ver que "animais-fofos-1.jpg" veio mesmo estando offline. Porém, há uma indicação que diz que o arquivo veio do cache - "(from disk cache)":



The screenshot shows a web browser's developer tools with the Network tab selected. A yellow box highlights a small, fluffy white dog on a wooden floor. Below the browser window, the Network tab is filtered to show only image requests. The table below lists the requests:

Name	Status	Type	Initiator	Size	Time	Timeline - Start Time
animais-fofos-1.jpg	200	jpeg	jquery.js:5322	(from disk cache)	1 ms	
bin2.svg	200	svg+xml	(index):-Infinity	(from disk cache)	2 ms	
edit.svg	200	svg+xml	(index):-Infinity	(from disk cache)	1 ms	

A imagem, portanto, já havia sido armazenada e, para que não fosse preciso consultar o servidor de novo, o que já existia no cache está sendo reutilizado. Este comportamento é algo que já tínhamos antes mesmo do Application Cache, significa que a imagem está "cacheada". O problema é que não sabemos por quanto tempo.

Dentro dos "Headers", disponível após o clique no nome do arquivo da imagem, nosso "Response Header" (as informações da resposta, em tradução do inglês), indica que, no campo "Cache-Control", a idade máxima desse cache é de 86400. Isto quer dizer que a tal imagem permanecerá cacheada por 86400s no computador, o que dá cerca de 24h, exatamente 1 dia. Após esse período, a imagem não estará mais no cache, e se estivermos offline, nosso browser não conseguirá pegar a imagem. Não preciso esperar a passagem desse tempo para testar isso, pois podemos recarregar a página removendo a imagem do cache.

Isso pode ser feito através do atalho "Ctrl + F5" caso esteja no Windows, ou "Command + Shift + R" no Mac. A imagem, então, deixa de ser carregada:

Se a imagem não está mais no cache, significa que precisamos adicioná-la ali. Já sabemos que no `localStorage` apenas o conteúdo url está armazenado. Se quero armazenar o que vem do servidor, preciso colocá-lo no Application Cache. Fazer isto significa que preciso ir até o `offline.manifest` e colocar o endereço da imagem, copiando e colando-o da barra do navegador, como fizemos anteriormente.

Já vimos essa história antes! Se fizermos isto, estaremos cacheando apenas essa imagem específica, não qualquer uma encontrada na internet, e não temos controle sobre o que o usuário pode escolher colocar na página. Precisamos de uma maneira que sirva quando o usuário for à nossa app, adicionar uma imagem qualquer no conteúdo do cartão, sendo cacheada no Application Cache. Ou seja, precisamos que a url dela seja armazenada dentro do Application Cache. Como fazer isto?

A app é utilizada através do navegador, e o Application Cache só funciona porque temos um arquivo carregado chamado `offline.manifest`, o qual lista tudo que deve estar cacheado, armazenado. O usuário precisa que a url da imagem desejada seja listada ali também. A única forma de o Application Cache mostrar algo é colocando a url no `offline.manifest`, que é o mesmo para todos os usuários. Imagine que qualquer pessoa que quiser colocar uma imagem como conteúdo do cartão precise ter a mesma url no mesmo arquivo, ou seja, todos os usuários teriam que adicionar a url das imagens de todos eles no mesmo `offline.manifest`.

A ideia, portanto, é tentar colocar um `offline.manifest` diferente para cada usuário... Isto significa que o `index.html` a ser carregado deve ser customizado, como por exemplo `offline-artur.manifest`. Teria que mudar o `index.html` para cada usuário que carregar o site. Seria trabalhoso demais e, além disso, o servidor a ser utilizado precisaria ser capacitado para tal.

Isto nos mostra que o Application Cache não foi feito para trabalhar com a ideia de ter um conteúdo dinâmico. O que você quer instalar são os arquivos `.css`, `.js` e o `index.html`. Porém, você também quer a possibilidade de colocar imagens no cache. Imagens que mudarão de usuário para usuário. Está aí um problema: não temos como cachear, de forma simples, pelo menos a partir de código no Front End, aquilo que queremos armazenar. Nos próximos vídeos, tentaremos resolver isso.

