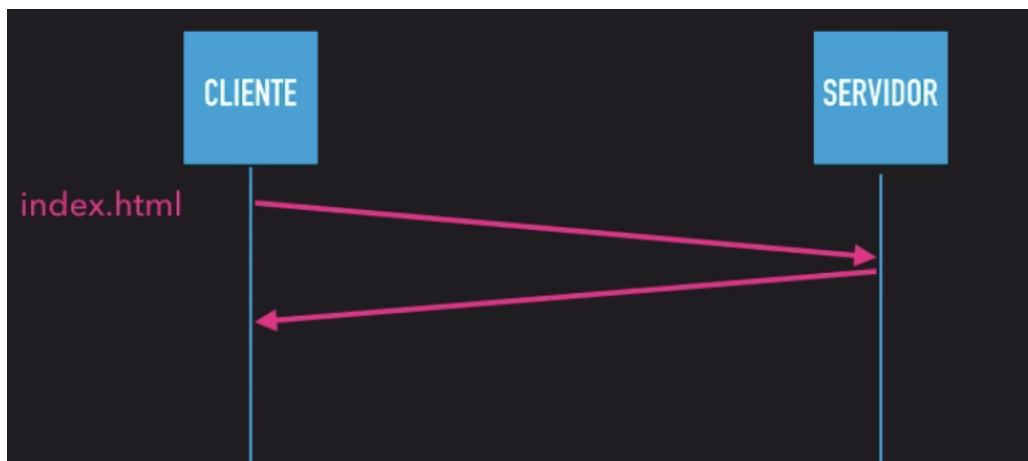


HTTP2 - Server Push

Transcrição

Temos o cliente e um servidor sendo representados. Podemos imaginar que estamos fazendo uma requisição para uma página principal, a **index.html**. Essa requisição bate no servidor e o servidor nos traz o conteúdo HTML.



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Caelum</title>
  <link rel="stylesheet" type="text/css" href="estilo.css">
  <script type="text/javascript" src="jquery.js"></script>
  <script type="text/javascript" src="principal.js"></script>
</head>
<body>
  ....
  </td>
</body>
</html>
```

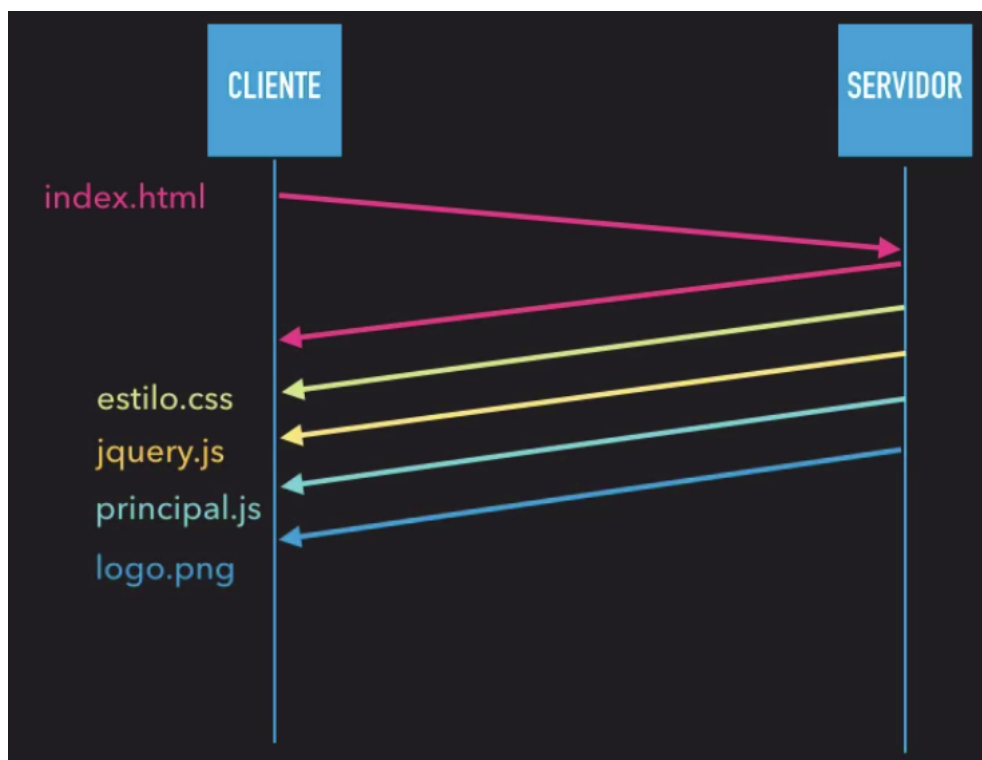
O HTML retornado pode ter o título **Caelum**, e então vai aparecer no nosso navegador essa informação. Além disso, temos um arquivo CSS, de estilização da página, que é o **estilo.css**, e dois arquivos JavaScript necessários para a página ser executada, o **jquery.js** e o **principal.js**. Além disso, no meio do corpo do HTML, tem um recurso que é de imagem, temos a imagem **logo.png**. Mas além desses, podemos ter vários outros recursos na nossa página.

Então, ao receber esse conteúdo, o browser tem que sair fazendo requisições de tudo o que é necessário para que ele renderize a página. O navegador interpreta esse conteúdo HTML de cima pra baixo, verifica que o primeiro recurso necessário é o **estilo.css**, aí ele vai lá buscar. O segundo recurso necessário, **jquery.js**, que é uma biblioteca JavaScript. E além disso, precisamos do **principal.js** e do **logo.png** :



Todos esses recursos especificados no HTML são novas requisições que o browser precisa fazer, e nosso cliente precisa executar. O servidor vai recebendo essas requisições, mas o cliente fica ali esperando até que essas respostas cheguem e o nosso browser consiga de fato renderizar o conteúdo para o usuário. Então há uma espera até essas respostas chegarem de fato, pois o servidor devolve as respostas das requisições na mesma sequência que foram geradas.

A partir do HTTP2, isso ficou um pouco diferente. Agora temos uma conversa mais paralela. Anteriormente estávamos trabalhando com conceitos de requisições seriais, fazíamos uma requisição e esperávamos receber, fazíamos outra requisição e esperávamos receber e por aí vai. No HTTP2, quando o cliente realiza uma requisição para `*index.html`, o servidor devolve a página, mas ele já pode passar para o browser as informações necessárias para que essa página possa ser, de fato, exibida. Ou seja, ele consegue dar um passo além:



Isso é uma outra abordagem que surgiu no HTTP2, muito mais interessante. Mas quando o browser for interpretar essa página HTML, vai ter que passar pelo conteúdo que especifica o arquivo CSS? Sim, mas quando ele passar pelo

`estilo.css`, vai verificar que já recebeu. Ou seja, ele percebe que já recebeu essas informações.

Este é o conceito de ***Server Push***, ou seja, o ***server*** envia dados para o cliente sem que o cliente tenha solicitado, tornando o tráfego de dados muito mais otimizado.