

## HTTP 2.0 - Tópicos da aula

- Evolução HTTP 0.9 até 1.1
  - HTTP 1.0 - ano 1996
  - HTTP 1.1 - RFC 2616 (1999) e RFCs 7230 a 7235 (junho/2014)
  - SPDY - ano 2009
  - HTTP 2.0 - RFC 7540 (maio/2015)
- HTTP 1.1 provê:
  - caching
  - suporte a proxy web
  - conexões persistentes: múltiplas requisições e respostas por conexão
  - autenticação
  - transferência chunked
- Pipelining e bloqueio (*head of line blocking*) . Limitações do pipelining.
- Múltiplas conexões TCP (2 a 8 por domínio, dependendo do navegador). Vantagens e desvantagens de múltiplas conexões TCP.
  - Início lento (*slow start*) do TCP (controle de congestionamento)
- Conexões persistentes. Codificação *chunked*.
- Otimizações para conteúdo web com HTTP 1.1
  - Reduzir chamadas ao DNS
  - Diminuir o número de requisições
  - Rede de Distribuição de Conteúdo (CDN)
  - Melhorar o cache com cabeçalhos *expire* e *etag*
  - *Negociar compactação de dados* (gzip)
  - Evitar redirecionamentos
  - Usar vários domínios para aumentar o número de conexões TCP (fragmentação de domínio)
  - Concatenação de objetos (scripts javascript, CSS), *spriting* de imagens
  - *Inlining* (embutir *scripts* e outros conteúdos nas páginas HTML)
  - Vantagens e desvantagens dessas otimizações
- HTTP 2.0 (maio/2015)
  - Google SPDY foi precursor
  - Não requer modificação das aplicações web
  - Aplicações podem usar recursos do HTTP 2.0 para melhorar a eficiência
  - Multiplexação da conexão TCP (várias requisições e respostas enviadas de forma concorrente no mesmo canal TCP) → vários fluxos concorrentes de dados sobre a mesma conexão TCP.
  - Permite a utilização de menos conexões TCP persistentes. Lembre-se: o TCP é otimizado para mensagens longas (controle de congestionamento)
  - Enquadramento binário. Divide e encapsula as mensagens em quadros. (discussão sobre protocolos de texto versus binários)
  - Mesmo formato das entidades de requisições e respostas e mesma semântica, mas muda a forma de utilização das conexões. Portanto, a especificação do HTTP 1.1 não se tornou obsoleta com a publicação da especificação do HTTP 2.0 pois tratam de diferentes questões.
  - Frame HTTP 2.0: cabeçalho de frame e dados
  - Cabeçalhos compactados
  - Compressão de dados

- Atribuição de prioridades: pesos e precedência
- *Push* do servidor: armazenados em cache pelo cliente, reutilizados em diferentes páginas, multiplexados junto com outros recursos, priorizados pelo servidor.
- Reduz a necessidade de otimizações ad hoc.
- Diminui a carga nos servidores e clientes.

**Referências:**

- [Introdução a HTTP/2](#) no site [Web / Fundamentals](#) da Google
- [HTTP/2 - Wikipedia](#)
- Ilya Grigorik, *High-Performance Browser Networking*, Editora: O'Reilly, 2013. Disponível online [aqui](#).