HTTP/1.1 Keep Alive

HyperText Transfer Protocol

HTTP/1.0

- Open connection
- Issue GET
- Server closes connection after response

HTTP/1.0 speed

- Latency: 50ms
- Request size: 1 full segment
- Response size: 2 full segments (size of slow start window)
- Segment packetization delay: 10ms (request and response), full duplex
- Maximum open connections: 4

- Case 1: Single page, 230ms
    - SYN: 50ms, SYN/ACK: 50ms, ACK/request: 60ms, response: 70ms
- Case 2: Page that loads 2 images: 480ms
    - Setup: 100ms, request/response: 130ms
    - Setup: 100ms, request/response: 150ms

HTTP/1.0

- Open connection
- Issue GET
- Server closes connection after response
- *Opening many connections is slow*
- *Many transfers are small, doesn't let TCP window grow*

HTTP/1.1

- Added Connection header for requests
  - keep-alive: tells the server “please keep this connection open, I’ll request more”
  - close: tells the server to close the connection
  - Server can always ignore
- Added Connection header for responses
  - keep-alive: tells the client it’ll keep the connection open
  - close: tells the client it’s closing the connection
- Added Keep-Alive header for responses
  - Tells client how long the connection may be kept open

HTTP/1.1 speed

- Latency: 50ms
- Request size: 1 full segment
- Response size: 2 full segments (slow start window is 30 segments)
- Segment packetization delay: 1ms (request and response)
- Maximum open connections: 2
- Page that loads 11 images

- HTTP/1.0 speed: 1,421ms
  - Page setup: 100ms, request/response: 103ms
  - 11 images. 6 x (image setup: 100ms + request/response: 103ms)
- HTTP/1.1 speed: 326ms
  - Connection setup: 100ms
  - Page request/response: 103ms
  - Image requests/responses: 123ms

SPDY

- Protocol proposed by Google to speed up the web
- Request pipelining
- Removes redundant headers
- Becoming basis of HTTP/2.0