

Jetzt bitte Terminal öffnen und überprüfen, ob eine Verbindung zum Internet steht.

# HTTP und HTTP/2

Michael Hochleitner  
30.11.2017

# Was ist HTTP?

- Hypertext Transfer Protocol
- Protokoll für Rechnernetze
- Senden von Datenpaketen
- Client und Server



HTTP spezifiziert die Gestalt der  
Datenpakete

Anfrage

GET /index.html

Methode Ressource

Antwort

```
<HTML>
A very simple HTML page
</HTML>
```

- Prototyp von 1991
- Textprotokoll
- Request-Response

# Grenzen von HTTP/0.9

- Schließen der TCP-Verbindung nach jeder Anfrage
- nur ein Dateiformat: HTML
- minimalistische Fehlerbehandlung

[https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics\\_of\\_HTTP/Evolution\\_of\\_HTTP](https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP)

# HTTP/1.1

- 1997
- erster offizieller Standard
- Status Codes
- Metadaten
- mehr Methoden
- Wiederverwendung von Verbindungen
- Pipelining

# Header

- Allgemeine Headerfelder
  - Date
- Anfrage Headerfelder
  - Accept
- Antwort Headerfelder
  - Location
- Entity Headerfelder
  - Content-type

<https://tools.ietf.org/html/rfc2616>

# Methoden

- GET
- HEAD
- POST
- PUT
- DELETE
- TRACE

<https://tools.ietf.org/html/rfc2616>



# Probleme mit HTTP/1.1

- Verwaltung von TCP-Verbindungen
- Ziel Nebenläufigkeit
- Pipelining vorhanden

- Head of Line Blocking

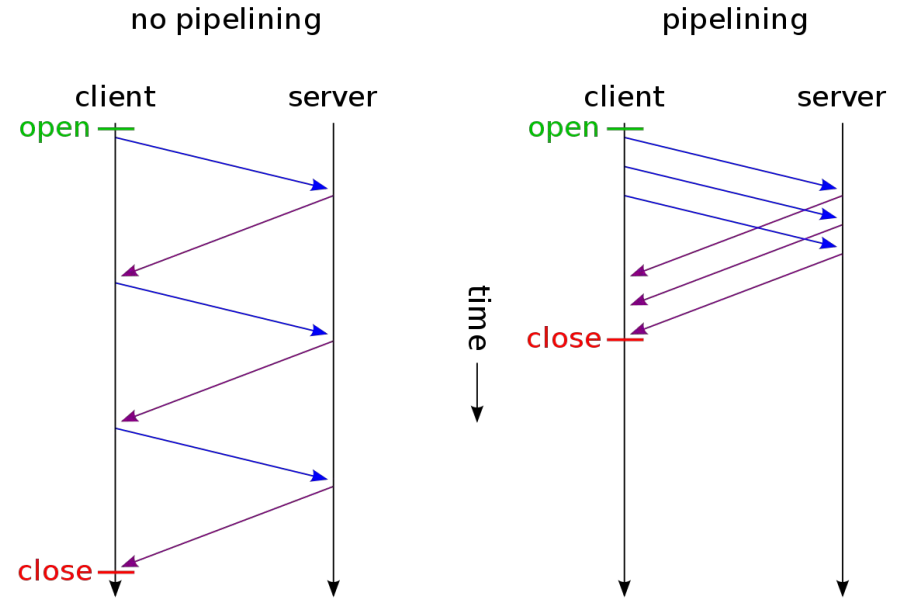


- viele Verbindungen für niedrige Latenz

<http://httpwg.org/specs/rfc7540.html>

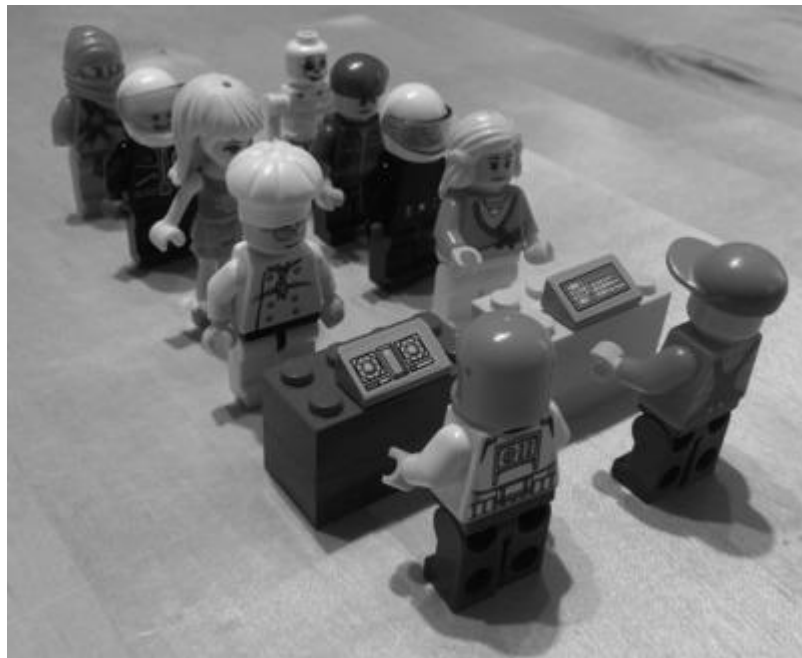
# Pipelining

- FIFO



# Head of Line Blocking

- Bearbeitungsdauer variiert
- Ungewissheit der Bearbeitungsdauer
- neue Reihe aufmachen möglich
- Skalierung begrenzt



<https://bagder.gitbooks.io/http2-explained/en/part2.html>

# HTTP/2

- 2015
- Beschleunigung der Übertragung
  - Datenkompression
  - Zusammenfassung von Anfragen
  - Server Push
  - binäre Kodierung von Inhalte

# Datenkompression

# Multiplexing

- Zusammenfassen von Anfragen

# Server Push

- Server initiierte Datenübertragung

# Binäre Kodierung von Inhalten

- HTTP/2 ist binär

<https://bagder.gitbooks.io/http2-explained>



# Zusammenfassung

- Netzwerkprotokoll
- verschiedene Versionen
- Fokus bei der Weiterentwicklung
  - Übertragungsgeschwindigkeit
  - Flexibilität

# Flexibilität

- Header
  - verschiedene Dateitypen
- verschiedene Funktionen
  - Semantik von Nachrichten
- Definition neuer Header möglich
- Definition neuer Methoden möglich

GET /en-US/docs/Glossary/Simple\_header HTTP/1.1

Host: developer.mozilla.org

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:50.0) Gecko/20100101 Firefox/50.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate, br

Referer: https://developer.mozilla.org/en-US/docs/Glossary/Simple\_header

- Versionsinformation
- Header
- Übertragung verschiedener Dateitypen

[https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics\\_of\\_HTTP/Evolution\\_of\\_HTTP](https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP)

200 OK

Connection: Keep-Alive

Content-Encoding: gzip

Content-Type: text/html; charset=utf-8

Date: Wed, 20 Jul 2016 10:55:30 GMT

Etag: "547fa7e369ef56031dd3bff2ace9fc0832eb251a"

Keep-Alive: timeout=5, max=1000

Last-Modified: Tue, 19 Jul 2016 00:59:33 GMT

Server: Apache

Transfer-Encoding: chunked

Vary: Cookie, Accept-Encoding

- Status Code
- Header

[https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics\\_of\\_HTTP/Evolution\\_of\\_HTTP](https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP)

