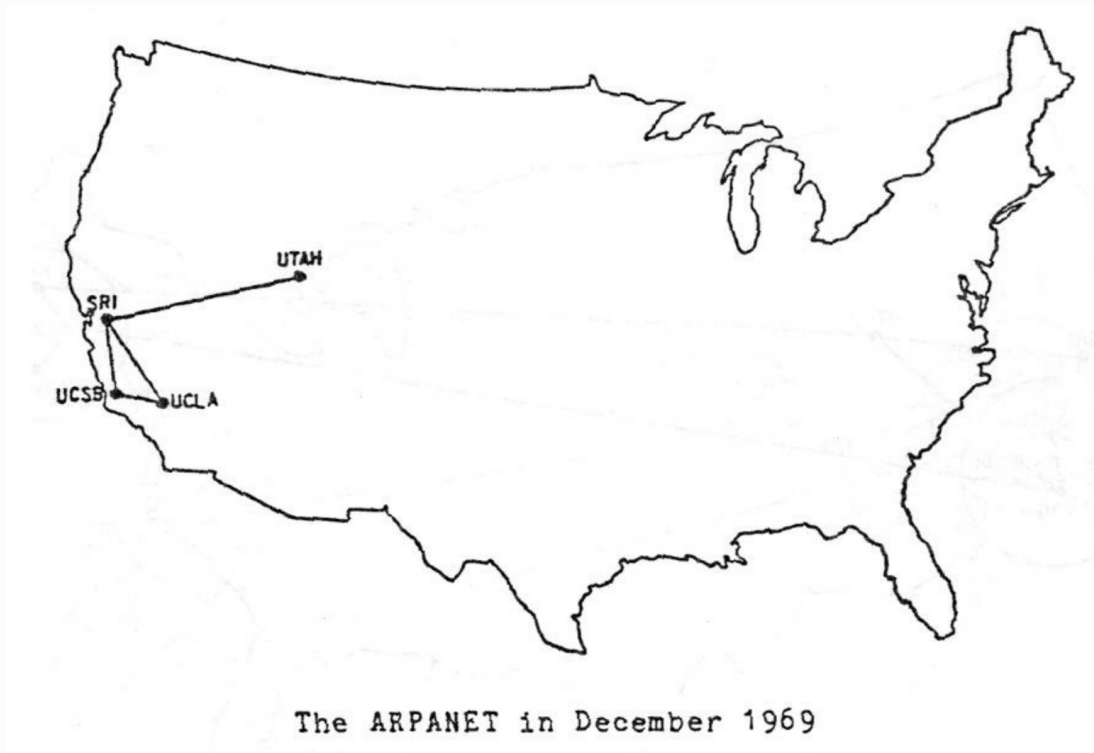


# Challenge 08: Web

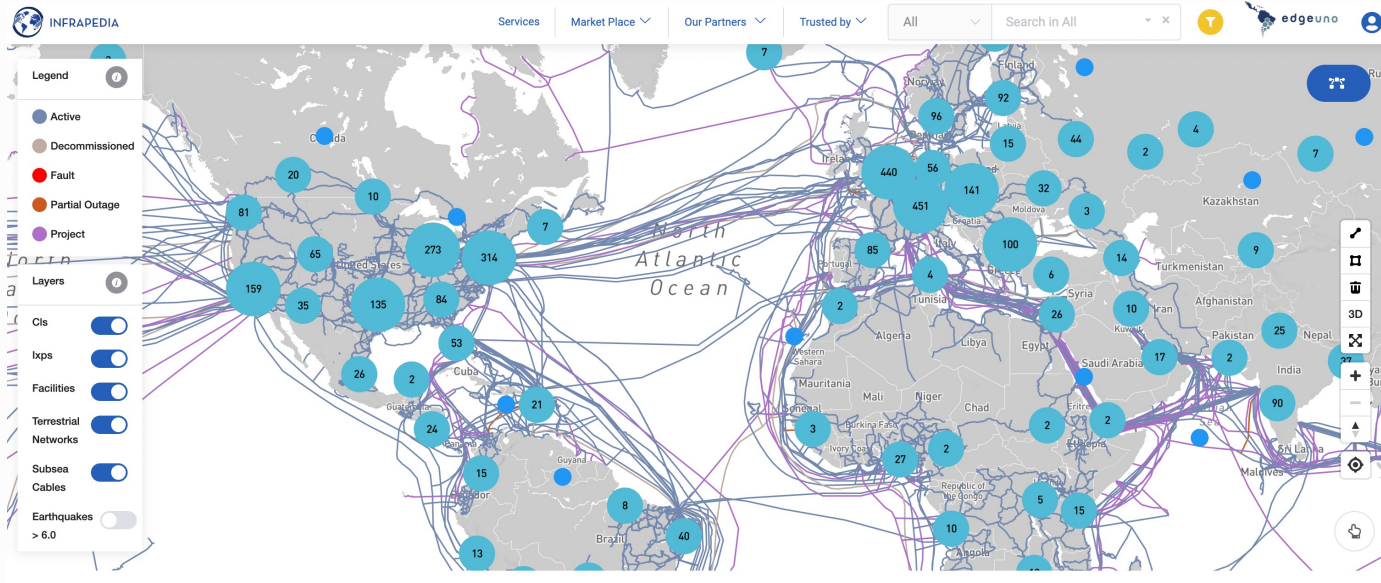
Internet

Prof. Dr. Markus Heckner

# Arpanet als erstes Netzwerk miteinander kommunizierender Rechner



# Heutiges Internet als Weiterentwicklung des Arpanet



Bildquelle: <https://www.infrapedia.com/app>

# Das Internet

- Verbindet alle per Kabel, WLAN oder Mobilfunk *angeschlossenen* Geräte
- **Router** leiten die Anfragen zwischen diesen Rechnern weiter
- **Protokolle** legen fest, wie Rechner miteinander kommunizieren
- Rechner im Internet kommunizieren über **TCP/IP**

Wie lassen sich Daten über das Internet verschicken?



# IP bestimmt die Adresse eines Rechners

# . # . # . #

142 . 250 . 186 . 131



TCP beschreibt welcher Dienst auf einem Rechner genutzt werden soll

80 HTTP

443 HTTPS

...



# IP Adressen kann man sich schlecht merken

Domain Name	IP-Adresse



URLs sind Adressen von Rechnern, die man in die Adresszeile eines Browsers eingeben kann

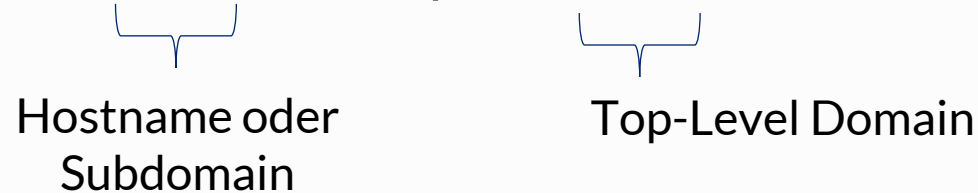
Protokoll                  Domain-Name



`http://www.example.com/`

URLs sind Adressen von Rechnern, die man in die Adresszeile eines Browsers eingeben kann

`http://www.example.com/`




Hostname oder  
Subdomain

Top-Level Domain

URLs sind Adressen von Rechnern, die man in die Adresszeile eines Browsers eingeben kann

*Fully-Qualified-Domain-Name*



http://www.example.com/

URLs sind Adressen von Rechnern, die man in die Adresszeile eines Browsers eingeben kann

`http://www.example.com/`



Fragt die Startseite  
einer Webseite an

URLs sind Adressen von Rechnern, die man in die Adresszeile eines Browsers eingeben kann

`http://www.example.com/folder/file.html`



Fragt eine Datei  
in einem Ordner ab

# Daten können per http über GET und POST übertragen werden

- GET – Teil der URL – Unsicher
- POST – Teil der eigentlichen Nachricht

# GET-Request

GET / HTTP/1.1

Host: www.example.com

} http-  
Header

# Response

HTTP/1.1 200 OK

Content-Type: text/html

...

}  
http-  
Header

}  
http-  
Body



# HTTP Statuscodes

- 200 OK
- 301 Moved Permanently
- 302 Found
- 304 Not Modified
- 307 Temporary Redirect
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found - Ein Klassiker
- 418 I'm a Teapot - Aprilscherz
- 500 Internal Server Error - Ausgelöst durch einen Bug im serverseitigen Code
- 503 Service Unavailable
- ...



# Challenge 08: Web

HTML

Prof. Dr. Markus Heckner

Welche Inhalte tauschen Browser und Server aus?



# HTML

- Elemente
- Tags
- Attribute

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">  
  <head>  
    <title>  
      hello, title  
    </title>  
  </head>  
  <body>  
    hello, body  
  </body>  
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```



```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

```
    hello, body
```

```
  </body>
```

```
</html>
```



```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <title>
```

```
      hello, title
```

```
    </title>
```

```
  </head>
```

```
  <body>
```

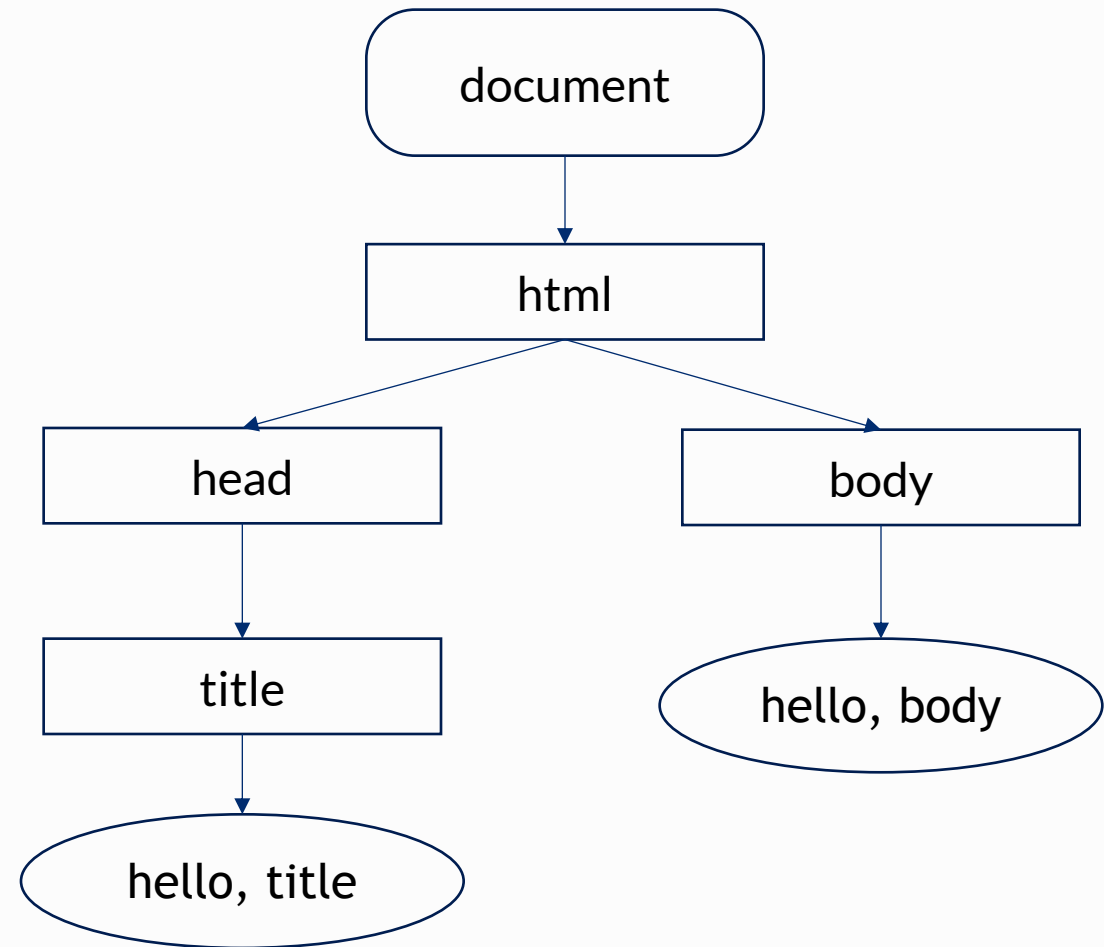
```
    hello, body
```

```
  </body>
```

```
</html>
```

```
<!DOCTYPE html>

<html lang="en">
  <head>
    <title>
      hello, title
    </title>
  </head>
  <body>
    hello, body
  </body>
</html>
```



# Daten übertragen - URL-Parameter mit GET

Pfad

`http://www.example.com/path`

# Daten übertragen - URL-Parameter mit GET

Schlüssel-  
Wert-Paar

  
`http://www.example.com/path?key=value`





# Challenge 08: Web

CSS

Prof. Dr. Markus Heckner

# CSS bestimmt Optik und Layout von Webseiten

```
body {  
    text-align: center;  
}
```

# Frameworks

- Stellen Code bereit, auf dem man aufsetzen kann
- Beschleunigen Entwicklung – Rad muss nicht immer neu erfunden werden
- Bootstrap beschleunigt Entwicklung des Frontends einer Webseite





# Challenge 08: Web

JavaScript

Prof. Dr. Markus Heckner

# Webseiten interaktiv gestalten mit JavaScript

```
let counter = 0;  
counter = counter + 1;  
counter += 1;  
counter++;
```

# JavaScript

```
if (x < y) {  
}
```



# JavaScript

```
if (x < y) {  
  } else {  
  }
```

# JavaScript

```
if (x < y) {  
  } else if (x > y) {  
  } else {  
  }
```

# JavaScript

```
while (true) {  
}
```



# JavaScript

```
for (let i = 0; i < 3; i++) {  
}
```

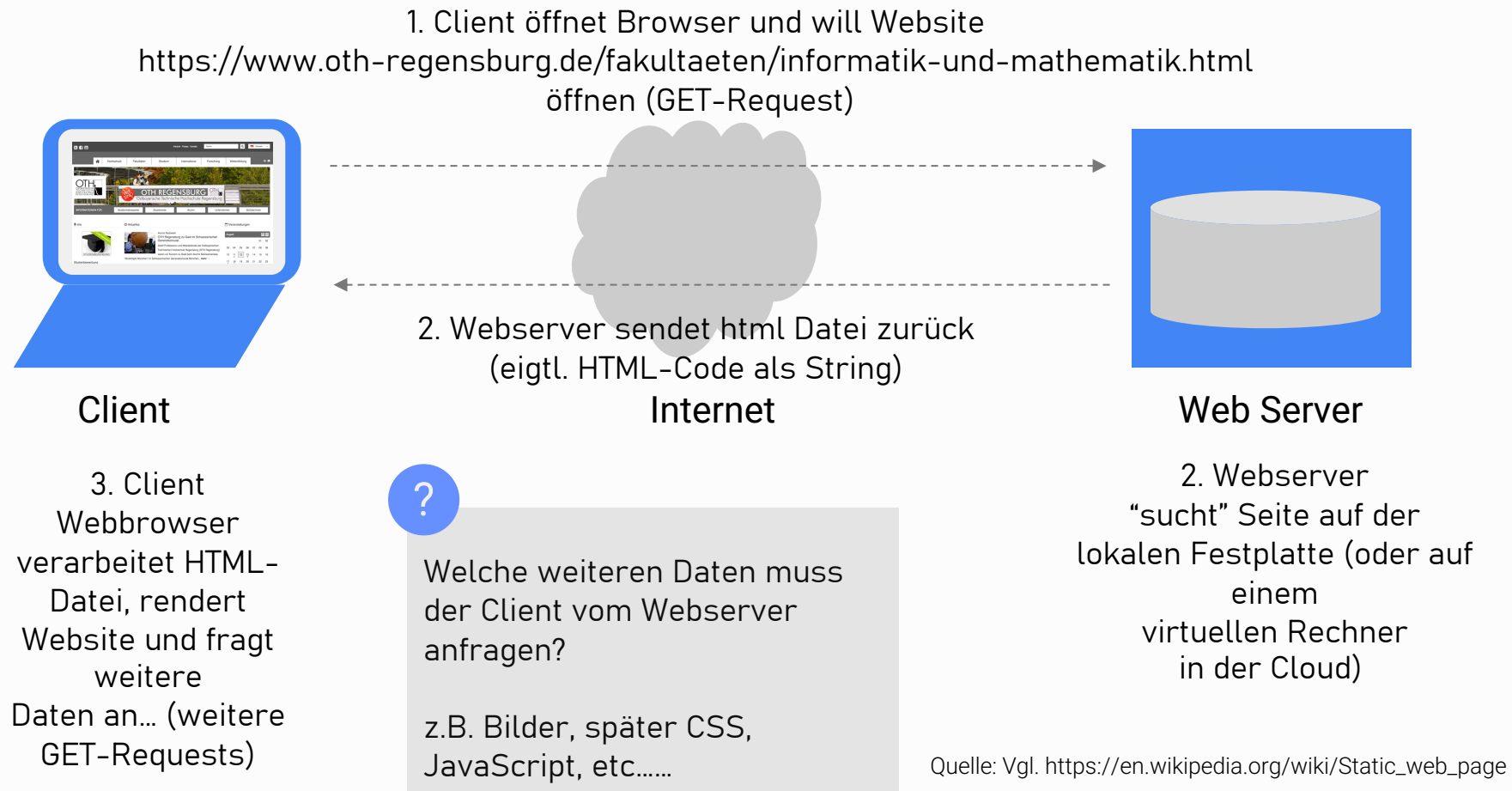






# ANHANG





9/9

0800 Antan started  
1000 stopped - antan ✓  
1300 (033) MP - MC 2.130476415 convt  
(033) PRO 2 2.130476415 convt  
Relays 6-2 in 033 failed speed speed test  
in relay 11,000 test.

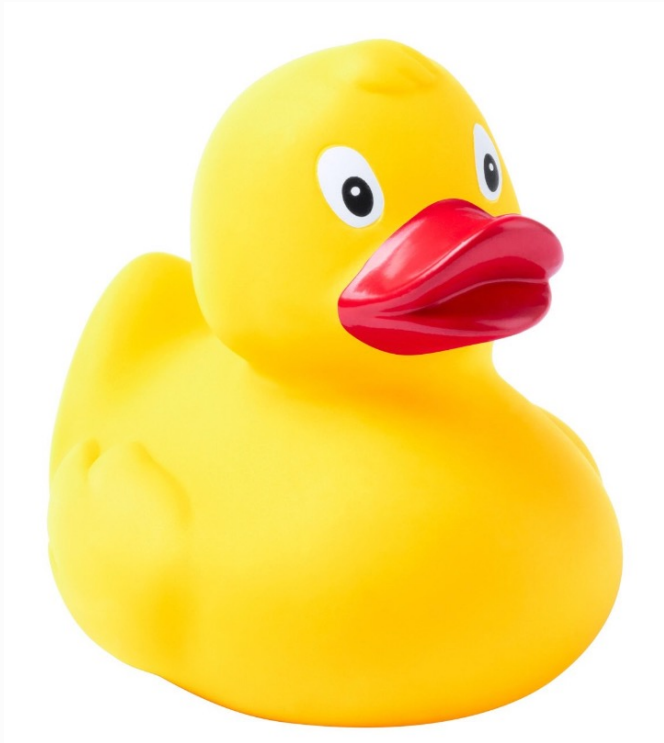
Relays changed  
Started Cosine Tape (Sine check)  
1525 Started Multi-Adder Test.

1545 Relay #70 Panel F (moth) in relay.

First actual case of bug being found.  
Antan started.  
1700 closed down.

Relay 3145  
Relay 327

# Quietscheenten-Debugging



Quelle: [https://en.wikipedia.org/wiki/Rubber\\_duck\\_debugging](https://en.wikipedia.org/wiki/Rubber_duck_debugging)

# Listen

## Wenn einzelne Variablen nicht reichen



Ablegen mehrerer Werte in einer einzelnen Variable, wie z.B.

- Tore in einer Bundesligasaison nach Spieltag
- Häufigkeiten aller Buchstaben eines Alphabets
- Noten aller Schüler einer Klasse
- ...



# Vertauschen von Variablenwerten

```
a = 5  
b = 3  
a = b  
b = a
```



```
print(f"Swapped variables a: {a} b: {b}")
```



# Objekte

- Objekte können mehrere Werte speichern (Liste ist ein Objekt)
- Objekte bündeln mehrere Funktionen in einer Variable





# Dictionaries als Nachschlagewerke

# Mehr zu Strings

- Strings sind Sequenzen aus einzelnen Buchstaben
- Strings sind Objekte
- Auslesen und Manipulieren von Strings ist ein häufiges Programmierproblem

# Parameter von der Shell

```
$ python3 argv0.py Markus  
hello, Markus
```



# Dokumentation

<https://docs.python.org/3/>

<https://docs.python.org/3/library/stdtypes.html#string-methods>

