

# **WBE: WEB-ENTWICKLUNG**

## **EINFÜHRUNG**

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

# WORUM GEHT'S?

- Web-Technologien...?
  - HTML, CSS, JavaScript, TypeScript
  - Canvas, SVG, Geolocation, DOM, Websockets, uvm.
- Web-Plattform...?
  - Web-APIs und deren Einsatzmöglichkeiten
  - z.B. Web-Apps, Electron-Apps, Webserver uvm.
- Web-Entwicklung...?
  - Tools, Frameworks, Libraries

Das ist ein riesiges Gebiet!

## Speaker notes

Curriculum Studium Informatik

<https://www.zhaw.ch/de/engineering/studium/bachelorstudium/informatik/curriculum-lehrplan/>

# ZIELE WBE

- JavaScript gut verstehen und einsetzen können
- Nebenbei: HTML und CSS verstehen (Selbststudium)
- Auswahl von Web-APIs kennen und einsetzen können
- Eigenes kleines Web-Framework erstellen können

Klar ist: Das Vorhaben ist immer noch sehr ambitioniert

## Speaker notes

- WBE wurde im HS20 zum ersten Mal durchgeführt
- Ersetzt drei Kurse: WEB1, WEB2, WEB3
- Komplett neu aufgebauter Kurs
- Seither jährlich mit kleineren Anpassungen

Falls noch nicht alles ganz rund läuft:

Beschwerden an [bkrt@zhaw.ch](mailto:bkrt@zhaw.ch) 😊

# ERWARTETE VORKENNTNISSE

- Grundkenntnisse HTML und CSS
  - Mail am Ende des letzten Semesters
  - Moodle-Kurs zum Selbststudium
  - In Tests und Praktikumsaufgaben vorausgesetzt
- Grundlegende Programmierkenntnisse
  - Programmieren 1/2
  - Software-Projekt 1/2

## Speaker notes

Der Vorbereitungskurs im Moodle enthält eine Menge an Informationen, mehr, als Sie für WBE benötigen. So können Sie selbst entscheiden, wie tief Sie sich in die Grundlagen von HTML und CSS einarbeiten möchten. Klar, je tiefer desto besser, da Web-Kenntnisse in vielen Informatik-Disziplinen von Vorteil sind. Der beim Vorbereitungskurs verlinkte [HTML & CSS Crash Course](#) (Tutorial auf Youtube) liefert in wenigen Stunden einen guten Überblick.

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

# ABLAUF

- Zwei Unterrichtslektionen pro Woche
- Zwei Praktikumslektionen pro Woche

Zeit	Art
100..120h	ca. Arbeitsaufwand 4 Credits (25..30h pro Credit)
42h	Präsenzlektionen ( $14 * 4 * 0.75$ )
58..78h	Selbststudium, Prüfungsvorbereitung, Prüfung

# KURSNOTE

Anteil	Art
20%	Leistungsnachweise im Semester
80%	Semesterendprüfung (SEP)

# LEISTUNGSNACHWEIS IM SEMESTER

- Kurztests
- Bewertete Abgaben im Praktikum

Leistungsnachweise im Semester können die Kursnote verbessern, nicht aber verschlechtern – für misslungene oder verpasste Leistungsnachweise wird die Note der SEP eingesetzt

Tipp: Nutzen Sie die Gelegenheit, sich durch die Leistungsnachweise im Semester ein Polster für die SEP zu verschaffen

## Speaker notes

Leistungsnachweise im Semester können die Kursnote verbessern, nicht aber verschlechtern. Leistungsnachweise, welche schlechter ausfallen als die SEP gehen mit der SEP-Note in die Bewertung ein. Verpasste Leistungsnachweise werden mit 1.0 bewertet.

Konsequenz: Für verpasste Leistungsnachweise wird die Note der SEP eingesetzt. Dafür gibt es für verpasste Leistungsnachweise keine Nachholtermine. Ausnahme: Krankmeldung mit Arztzeugnis.

# SEMESTERENDPRÜFUNG

- Mix aus verschiedenen Aufgaben zu den WBE-Themen
  - Multiple-Choice (vor allem Variante **Kprim**)
  - Fragen mit kurzen Antworten
  - Code-Aufgaben
- **Wir haben noch keine komplette Musterprüfung** 😬
- Tests können aber als Muster für die SEP angesehen werden

## MULTIPLE-CHOICE MIT KPRIM

Aus <https://teachingtools.uzh.ch/de/tools/multiple-choice-co-aufgaben-erstellen>:

*Der Aufgabentyp Kprim ist vor allem für Prüfungssettings interessant. Er besteht immer aus einer Vignette und vier Antwortalternativen, die jede für sich entweder richtig oder falsch sein kann. Dieser Typ hat sich testtheoretisch bewährt (gute Trennschärfe, geringe Ratewahrscheinlichkeit). Per Definition gibt es nur die volle Punktzahl, wenn alle vier Antwortalternativen richtig beurteilt wurden. Bei drei von vier korrekten Entscheidungen gibt es die halbe Punktzahl, alle anderen Lösungsmuster geben 0 Punkte.*

Dazu noch ein Hinweis: Wenn zu einer Antwortalternative keine Auswahl getroffen wird, wird dies als falsch gewertet. Es ist daher sinnvoll, alle Antwortalternativen zu beurteilen.

## MULTIPLE-CHOICE MIT UNTERPUNKTEN

Die Variante Kprim eignet sich vor allem, wenn Frage und Antwortmöglichkeiten einen engen Zusammenhang haben. Falls dies nicht der Fall ist, kann die Variante "Unterpunkte" eingesetzt werden, in der jede korrekte Antwort mit Teipunkten bewertet wird.

## ANDERE FRAGEN

- Fragen mit kurzen Antworten können zum Beispiel sein: „Welchen Wert liefert die folgende Funktion, wenn sie mit ... aufgerufen wird?“
- Code-Aufgaben wird es nur vereinzelt geben, zum Beispiel: „Schreiben Sie eine Funktion, welche...“

# UNTERLAGEN

In elektronischer Form (PDF) im Moodle:

- Die im Unterricht gezeigten Folien
- Ergänzende Hinweise zu den Folien (Lecture Notes)
- Praktikums-Aufgaben und Beispiel-Code

<https://moodle.zhaw.ch/course/view.php?id=25351>

# UNTERRICHT UND PRAKTIKUM

## Praktikumslektionen

- An den Notebooks gearbeitet
- Gespräche, Diskussionen, Teamarbeit erwünscht

## Unterrichtslektionen

- Vortrag und Demos, Diskussionen mit ganzer Klasse
- Gespräche untereinander bitte vermeiden – führen zu schlechterem Unterricht und stören alle
- Notizen machen empfohlen

## NOTEBOOKS IM UNTERRICHT

Students are Better Off without a Laptop in the Classroom

<https://www.scientificamerican.com/article/students-are-better-off-without-a-laptop-in-the-classroom/>

Bleibt natürlich jedem selbst überlassen...

## KURSVORGABEN

- Wichtige Regelungen in den Kursvorgaben (bitte lesen)
- Sollten Änderungen nötig sein, werden die Kursvorgaben angepasst
- Das wird aber vorgängig in den Lektionen diskutiert

Bitte erkundigen Sie sich bei den anderen Kursteilnehmern, wenn Sie einmal nicht anwesend sein können

## PRÜFUNGSRELEVANTER STOFF

Das ist nicht immer ganz klar abgrenzbar. Wenn Sie den Stoff im Wesentlichen verstanden haben, sollte eine genügende Note erreichbar sein, wenn Sie die Themen gut beherrschen eine gute Note, und für eine 6 ist die Erwartung, dass Sie sich über den vorgegebenen Stoff hinaus mit den Themen auseinandersetzen.

Der behandelte Stoff umfasst:

- Die geforderten Vorkenntnisse
- Die behandelten Themen
  - Material im Moodle
  - Erklärungen dazu in den Lektionen
  - Lösen der Praktikumsaufgaben
  - Aufgaben fürs Selbststudium

# Fragen?



# HINWEIS

Zum Einstieg in die WBE-Themen sind in den folgenden Abschnitten einige Eckpunkte der Web-Grundlagen noch einmal zusammengestellt

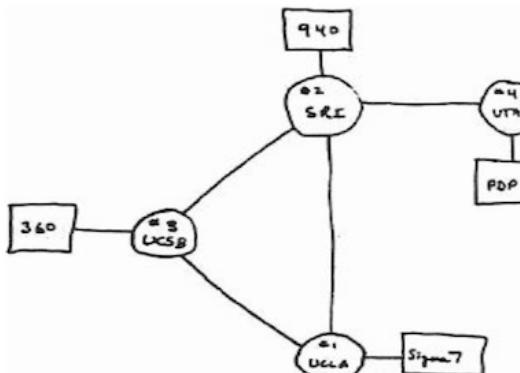
Diese sind Gegenstand des Vorbereitungskurses (Moodle), wo Sie weiteres Material zu diesen Themen finden

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

# INTERNET

- Weltweites Netzwerk, bestehend aus vielen Rechnernetzwerken
- Ursprünglich: ARPANET (1969: vier Knoten)
- Als Internet ab 1987 bezeichnet (ca. 27 000 Knoten)



THE ARPA NETWORK

DEC 1969

4 NODES

Dienste im Internet:  
E-Mail, Usenet, S/FTP, World Wide Web,...

# WORLD WIDE WEB

- Service, der auf dem Internet aufbaut
- Tim Berners-Lee  
(Anfang der 90er Jahre am CERN in Genf auf NeXT Computer)



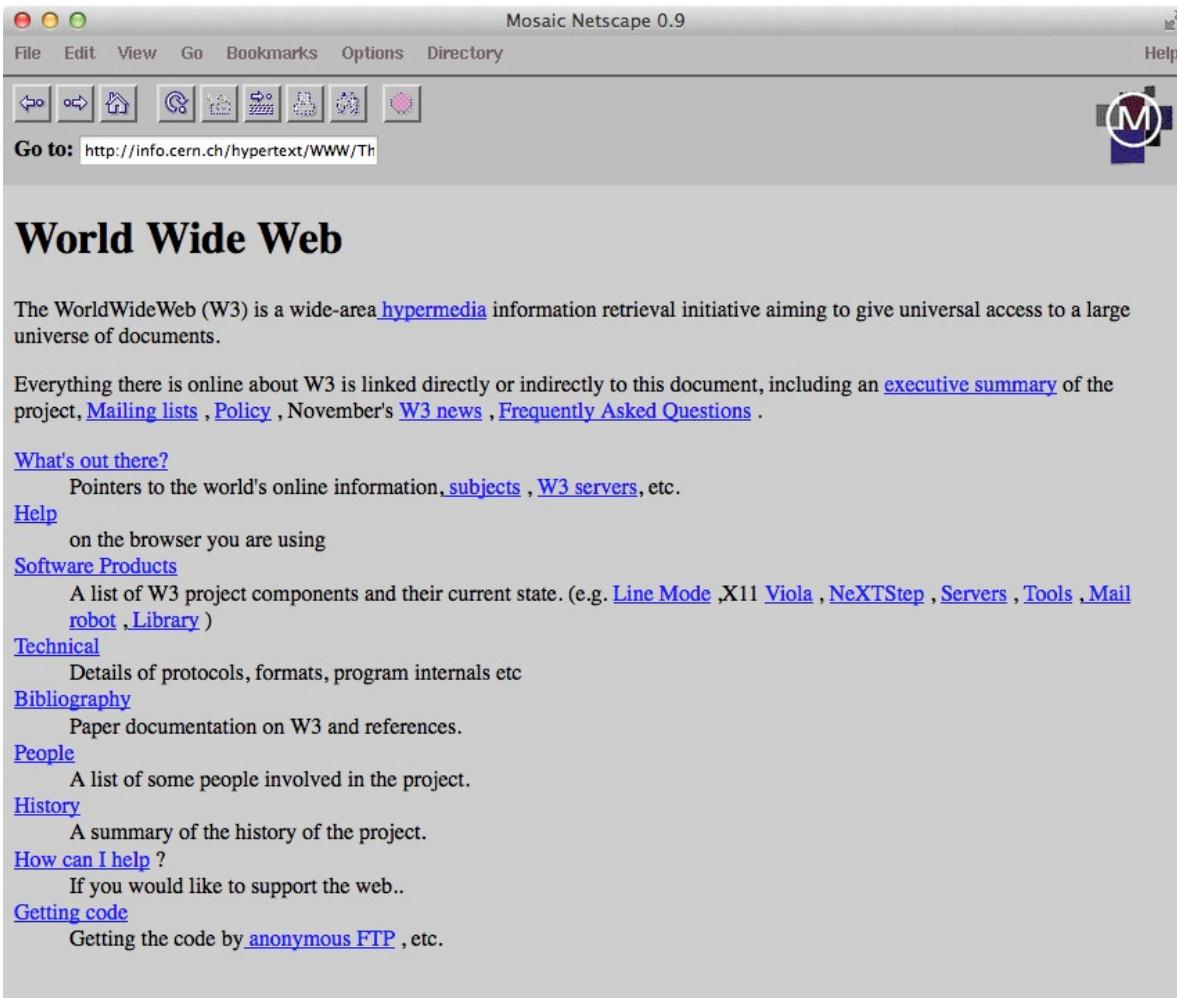
Tim's erster Webserver

Ankündigung des WWW im Usenet

## Speaker notes

Das World Wide Web (WWW) wird heute meist einfach als das *Web* bezeichnet.

# WEB-GESCHICHTE 1992



The screenshot shows the Mosaic Netscape 0.9 browser window. The title bar reads "Mosaic Netscape 0.9". The menu bar includes "File", "Edit", "View", "Go", "Bookmarks", "Options", "Directory", and "Help". Below the menu is a toolbar with icons for back, forward, search, and file operations. The address bar shows the URL "http://info.cern.ch/hypertext/WWW/Th". The main content area displays the text "World Wide Web" and a brief description of the W3 project, mentioning hypermedia, executive summary, mailing lists, policy, news, and frequently asked questions. A sidebar on the left lists various links such as "What's out there?", "Help", "Software Products", "Technical", "Bibliography", "People", "History", "How can I help?", and "Getting code".

**World Wide Web**

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

[What's out there?](#)  
Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

[Help](#)  
on the browser you are using

[Software Products](#)  
A list of W3 project components and their current state. (e.g. [Line Mode](#) ,[X11 Viola](#) ,[NeXTStep](#) ,[Servers](#) ,[Tools](#) ,[Mail robot](#) ,[Library](#) )

[Technical](#)  
Details of protocols, formats, program internals etc

[Bibliography](#)  
Paper documentation on W3 and references.

[People](#)  
A list of some people involved in the project.

[History](#)  
A summary of the history of the project.

[How can I help?](#)  
If you would like to support the web..

[Getting code](#)  
Getting the code by [anonymous FTP](#) , etc.

The birth of the Web  
(CERN)

Ankündigung NCSA  
Mosaic

# WEB-GESCHICHTE 1996...



## Speaker notes

# WAYBACKMACHINE

Alte Websites ansehen ist möglich mit der WayBackMachine:

<https://archive.org>

Altavista 1996

The screenshot shows the Altavista homepage from 1996. At the top, there's a navigation bar with icons for back, forward, search, and other functions. The URL in the address bar is <http://www.altavista.com/>. The main header features the Altavista logo with the tagline "Technology, Inc." and "View Multimedia From Our Vantage Point". Below the header is a search bar containing the text "AutoByTel". A link below the search bar says "Click here for advertising information - reach millions every month!". The search form includes dropdown menus for "Search the Web" and "Display the Results", and a "Submit" button. Below the search form is a link to "Search with Digital's Alta Vista". There are two buttons at the bottom left: one for "Contests" with the text "Make Me Laugh..." and another for "Creative Web" with the text "Create a Site...". At the bottom center, there's a link to "Download free demo versions of AltaVista Technology software". The footer contains the Altavista logo again and a horizontal line followed by the text "[Creative][Search][Humor][Email]".

The screenshot shows a web browser window with the URL <http://www.twi.ch/>. On the left, the red 'TWI' logo is displayed above the text 'TECHNIKUM WINTERTHUR' and 'INGENIEURSCHULE DES KANTONS ZÜRICH'. Below this is a small blue shield emblem. To the right of the logo is a vertical column of seven images, each followed by a colored square and a link. The images show various engineering and industrial scenes: a landscape, a close-up of machinery, a metal part being machined, a white fibrous material, two test tubes, and a close-up of a mechanical component.

-  [Allgemeine Informationen und Aktualitäten](#)
-  [Abteilungen / Diplomstudien](#)
-  [Nachdiplomstudien / Weiterbildung](#)
-  [Forschung und Entwicklung / Dienstleistungen](#)
-  [Zentrale Dienste \(Mensa ... Informatik\)](#)
-  [Personen und schulinterne Organisationen](#)
-  [Partnerorganisationen](#)
-  [WWW-Werkzeuge und Startseiten](#)

last modified: Sunday September 21 1997 14:30 MET DST by TWI Webmaster

# Netscape 1997

<https://web.archive.org/web/19970220012313/http://home.netscape.com:80/>

The screenshot shows the Netscape homepage from July 9, 1997. At the top, there's a banner with the text "WELCOME TO NETSCAPE" and "YOUR GUIDE TO NAVIGATING THE NET". Below the banner is a menu bar with links: GUIDE TO INTERNET, COMPANY & PRODUCTS, ENTERPRISE SOLUTIONS, DEVELOPERS, NETSCAPE STORE, ASSISTANCE, and ONE STOP SOFTWARE. A large headline "200 LEADING COMPANIES CHOOSE NETSCAPE" is prominently displayed. On the left, a news item dated JULY 9, 1997, discusses Netscape surpassing 2 million seats. A list of news items follows, including the release of Communicator 4.01 and the developer release of Java Foundation Classes. To the right, there's a section titled "GET THE LATEST NETSCAPE SOFTWARE" with links to Tune Up to Communicator, Get Any Netscape Software, and Netscape Now for Subscribers Only. Below that is a "NETSCAPE SPONSORS" section with a placeholder box. Further down, a "NETSCAPE STORE SPECIAL" section offers savings on FastTrack and LiveWire Servers, and Communicator 4.01. The bottom of the page features a "FOCUS ON SOLUTIONS" section with four columns: In-Box Direct, ISP Select, Product Update, and In the Enterprise. At the very bottom, there are links for visiting the international site, today's tip, powerstart, feedback, net search, and copyright information.

WELCOME TO  
NETSCAPE  
YOUR GUIDE TO NAVIGATING THE NET

GUIDE TO INTERNET COMPANY & PRODUCTS ENTERPRISE SOLUTIONS DEVELOPERS NETSCAPE STORE ASSISTANCE ONE STOP SOFTWARE

**200 LEADING COMPANIES CHOOSE NETSCAPE**

**JULY 9, 1997**

**N**etscape surpasses a key milestone of [200 design wins](#) with Netscape Communicator and SuiteSpot 3.0 software - now reaching more than 2 million seats.

- Download Netscape [Communicator 4.01](#) for Macintosh and Windows. Communicator Preview Release 6 is now [available](#) for Unix.
- To simplify the development of Internet and intranet applications, Netscape and Sun [announce](#) the developer release of Java Foundation Classes.
- The beta version of Netscape [Messaging Server 3.0](#) for Windows NT is now [available](#).
- Get the *New York Times*, SportsLine, and HotWired delivered daily via [Netscape In-Box Direct](#).
- [More news ...](#)

**HEADLINES FROM ABCNEWS.COM**

**ABC NEWS** Dover Roams Mars

**NETSCAPE COLUMNS**

In this week's [The Main Thing](#) column, Jim Barksdale talks about team collaboration. Read all the [Netscape Columns](#) to keep up with Internet issues and trends.

**FOCUS ON SOLUTIONS**

**IN-BOX DIRECT** Subscribe to [Netscape In-Box Direct](#) and receive free content from NewsHound, E! Online, and National Geographic.

**ISP SELECT** Get 30 days of [free Internet access](#) when you sign up with a participating Local Bell Internet Service Provider through Netscape ISP Select.

**PRODUCT UPDATE** Netscape [SuiteSpot 3.0](#) delivers an integrated suite of server software for web, messaging, and crossware applications.

**IN THE ENTERPRISE** Marc Andreessen outlines Netscape's future vision and new software designed to make the [Networked Enterprise](#) a reality.

Visit a Netscape International Site Today's Tip | PowerStart | Feedback | Net Search

NETSCAPE HOME DOWNLOAD SOFTWARE CUSTOMER SERVICE TECHNICAL SUPPORT SEARCH & CONTENTS WEB SITE ADVERTISING

Corporate Sales: 415/937-2555; Personal Sales: 415/937-3777; Government Sales: 415/937-3678  
If you have any questions, please visit [Customer Service](#), or contact your nearest [sales office](#).

Copyright © 1997 Netscape Communications Corporation  
This site powered by [Netscape SuiteSpot](#) servers.

## WEB WIE FRÜHER...

Wer das Web wie früher erleben möchte, kann <https://theoldnet.com/docs/httpproxy/index.html> aufrufen. Durch Setzen von *theoldnet.com* als Proxy mit der Jahreszahl als Port (1996 - 2012) kann man im Web des entsprechenden Jahres surfen. Wozu ist das gut? Auch hier gilt: Je besser man die Vergangenheit kennt, desto besser wird man die Zukunft meistern.

# WEB HEUTE

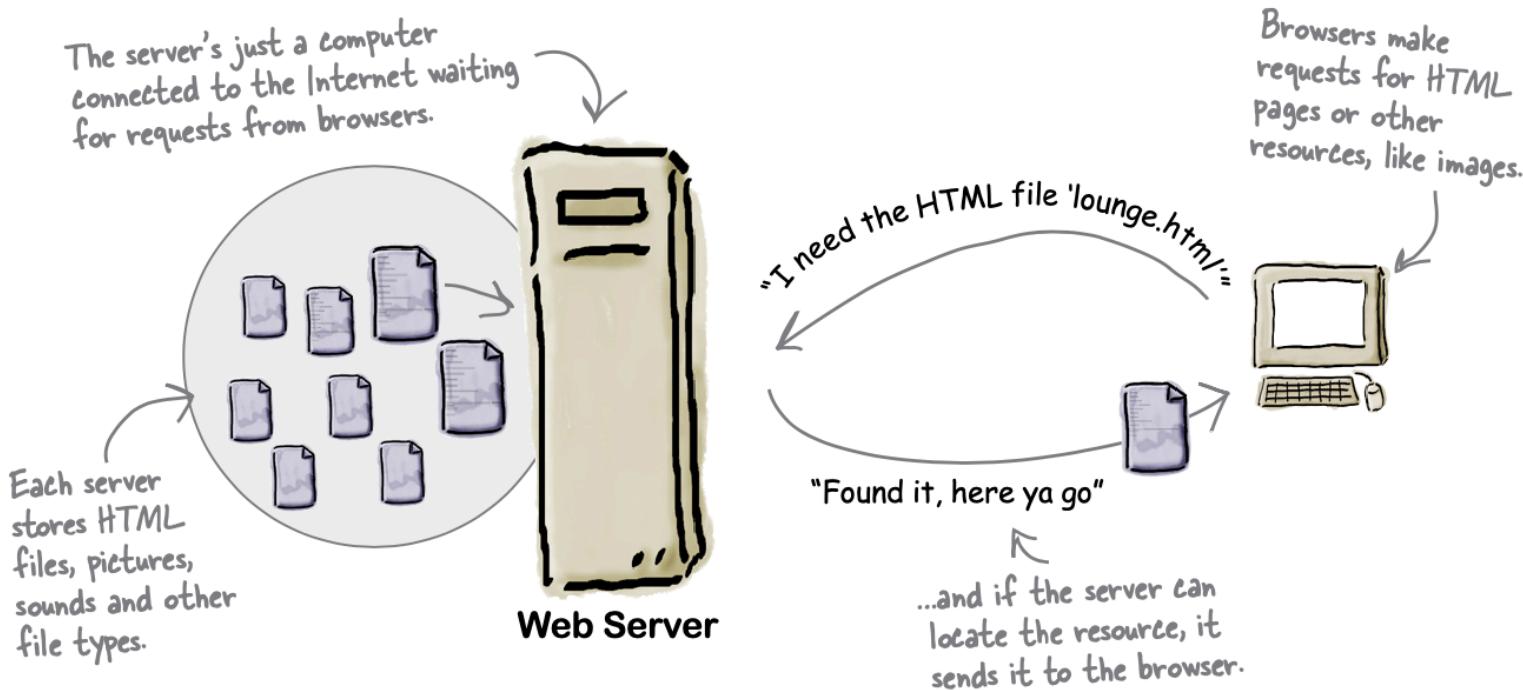
- Wichtige Applikations- und Informationsplattform
- Unzählige Technologien und Spezifikationen
- Vieles davon rund um „HTML Living Standard“
- Zahlreiche weitere APIs und Standards
- Grundlagen: HTML + CSS + JavaScript

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

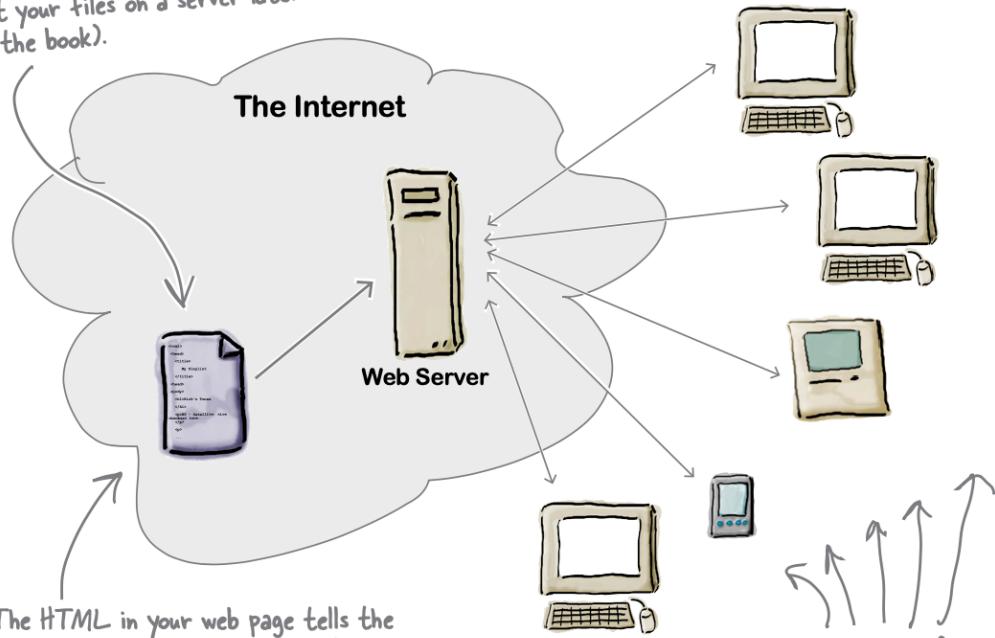
# WEB-ARCHITEKTUR

- Client-Server
- Interaktion: Request/Response



## Speaker notes

To make web pages, you create files written in the HyperText Markup Language (HTML for short) and place them on a web server (we'll talk about how to get your files on a server later in the book).



Once you've put your files on a web server, any browser can retrieve your web pages over the Internet.

The HTML in your web page tells the browser what it needs to know to display your page. And, if you've done your job well, your pages will even display well on cell phones and mobile devices, and work with speech browsers and screen magnifiers for the visually impaired.

And there are a lot of PCs and devices connected to the Internet all running web browsers. More importantly, there are friends, family, fans, and potential customers using those devices!

# CLIENT: BROWSER / USER AGENT



Firefox



Safari



Chromium



Microsoft Edge



Firefox Developer Edition



Safari Technology Preview



SeaMonkey



Vivaldi



<https://gs.statcounter.com>

## Speaker notes

Solche Angaben sind immer mit Vorsicht zu genießen. Eigentlich müsste genauer geschaut werden, wie die Daten erhoben wurden. Wenn zum Beispiel die Zugriffe auf eine bestimmte Menge von Websites ausgewertet werden, wird das Ergebnis durch Art und Inhalte der ausgewählten Websites beeinflusst.

Sehr stark hängen die Anteile auch von der ausgewerteten Region ab. Weltweit hat Chrome zum Beispiel einen höheren Marktanteil als in der Schweiz.

Auf Mobilgeräte beschränkt hat in der Schweiz aufgrund der starken Stellung von iOS der Safari einen guten Marktanteil:



# USER AGENT: LYNX

The screenshot shows a terminal window titled "Schreibtisch — lynx — 91x24" running the Lynx text-based web browser. The title bar also displays "LYNX – The Text Web-Browser (p1 of 2)". The content area shows the homepage of <http://lynx.invisible-island.net/>. The page includes copyright information from 1997-2017, 2018 by Thomas E. Dickey, and a menu with links to the home page, current development, stable release, and resources. A link to "LYNX – The Text Web-Browser" is present. The text "Lynx is the text web browser." is displayed. Below it, the text "This is the toplevel page for the Lynx software distribution site." is shown. Further down, it says "The current development sources have the latest version of Lynx available (development towards 2.9.0)." and "The main help page for lynx-current is online; the current User Guide is part of". At the bottom, there is a blue status bar with the text "-- Leertaste für nächste Seite --", keyboard navigation instructions ("Pfeile: Auf/Ab: andere Seite im Text. Rechts: Verweis folgen; Links: zurück."), and a help menu ("Hilfe O)ptionen P) Druck G)ehe zu M) Hauptseite Q) Beenden /=Suche <=History").

## Speaker notes

Bei der Bezeichnung *Browser* denkt man zuerst einmal an einen grafischen Browser wie Chrome oder Firefox. Das World Wide Web Consortium (W3C) bevorzugt statt *Browser* die Bezeichnung *User Agent*, um klarzustellen, dass es ganz unterschiedliche Arten von Clients gibt, mit denen man auf das Web zugreifen kann.

Hier einige Beispiele:

- Grafischer Browser auf Notebook (div. Bildschirmauflösungen,-größen)
- Grafischer Browser auf Smartphone (div. Bildschirmauflösungen,-größen)
- Grafischer Browser auf Tablet
- Einfacher Browser auf Feature Phone
- Sprachausgabe oder tastbare Braille-Ausgabe
- Browser auf Fernsehgerät
- Ausgedruckte Webseite
- Text-only-Browser
- Programm, das Webinhalte lädt und verarbeitet, z.B. Suchmaschine

Es kann durchaus nichts schaden, Ihre eigenen Seiten einmal mit dem *Lynx* anzusehen. Das Ergebnis ist etwa das, was auch Google auf Ihren Seiten „sieht“.

<https://lynx.invisible-island.net>

# BROWSER WARS

- Zunächst: Mosaic
- Später: Netscape
- Dann: Internet Explorer
- Heute: Chrome

Problematisch waren immer die Phasen, in denen ein Browser deutlich marktbeherrschend war

„This page is best viewed with Internet Explorer 6 on Windows.“

## Speaker notes

Wann immer ein Browser deutlich marktbeherrschend war, haben dessen Entwickler gerne neue Features in den Browser eingebaut, ohne Rücksicht auf deren Standardisierung zu nehmen. Dadurch entsteht leicht einen Tendenz weg vom offenen Web hin zu einem proprietären Client-Server-System.

Aktuell erleben wir wieder so eine Phase: Das W3C ist weitgehend entmachtet und nur noch eingeschränkt involviert in die weitere Entwicklung der Webstandards. Zudem dominiert der Browser Chrome die Landschaft der User Agents. Viele alternativen Browser bauen zudem auf der Rendering Engine Blink des Chromium-Projekts auf, auf dem auch der Chrome basiert.

Als Web-Entwickler sollten wir uns für eine möglichst grosse Browser-Vielfalt und eine gemeinsame und überlegte Weiterentwicklung der Webstandards einsetzen. Nachdem Microsoft für der Browser Edge die eigene Rendering Engine aufgegeben hat und ebenfalls auf Blink setzt, gibt es nur noch wenige Alternativen:

- Firefox: verschiedene Plattformen
- Safari: Apple-Plattformen
- Alle Browser unter iOS, da Apple vorgibt, dass nur Webkit als Rendering Engine unter iOS zulässig ist (auch nicht gerade im Sinne eines offenen Web und gerät zunehmend unter Druck)

# WEBSERVER

- Reagieren auf Anfragen von User Agents
- Senden diesen die angefragten Dateien
- Beispiele
  - Apache HTTP Server
  - Nginx (ausgesprochen „engine-ex“)
  - Microsoft Internet Information Services (IIS)
  - Apache Tomcat (Java)

# WEB-PLATTFORM

## Drei Grundpfeiler

- Sprachen: Rolle von HTML, CSS und JavaScript
- Adressierung: URI/URL, IP-Adresse, Domain-Name
- Protokoll: HTTP

Gleich mehr zu den Sprachen. Schauen wir zuerst kurz auf Adressierung und Protokoll...

# ADRESSIERUNG

## Uniform Resource Locator, URL

"http://" [user[:password]@] host [:port] path [ "?" query ] [ "#" fragment ]

scheme-specific-part → → → |  
|  
**http://hans:geheim@example.org:80/demo/example.cgi?land=de&stadt=aa#geschichte**  
| | | | | | | |  
| | | host | url-path | query | fragment  
| | password | port |  
| user |  
scheme (hier gleich Netzwerkprotokoll)

(Quelle: Wikipedia)

## Speaker notes

Uniform Resource Locators (URL) (RFC 1738)

<https://www.ietf.org/rfc/rfc1738.txt>

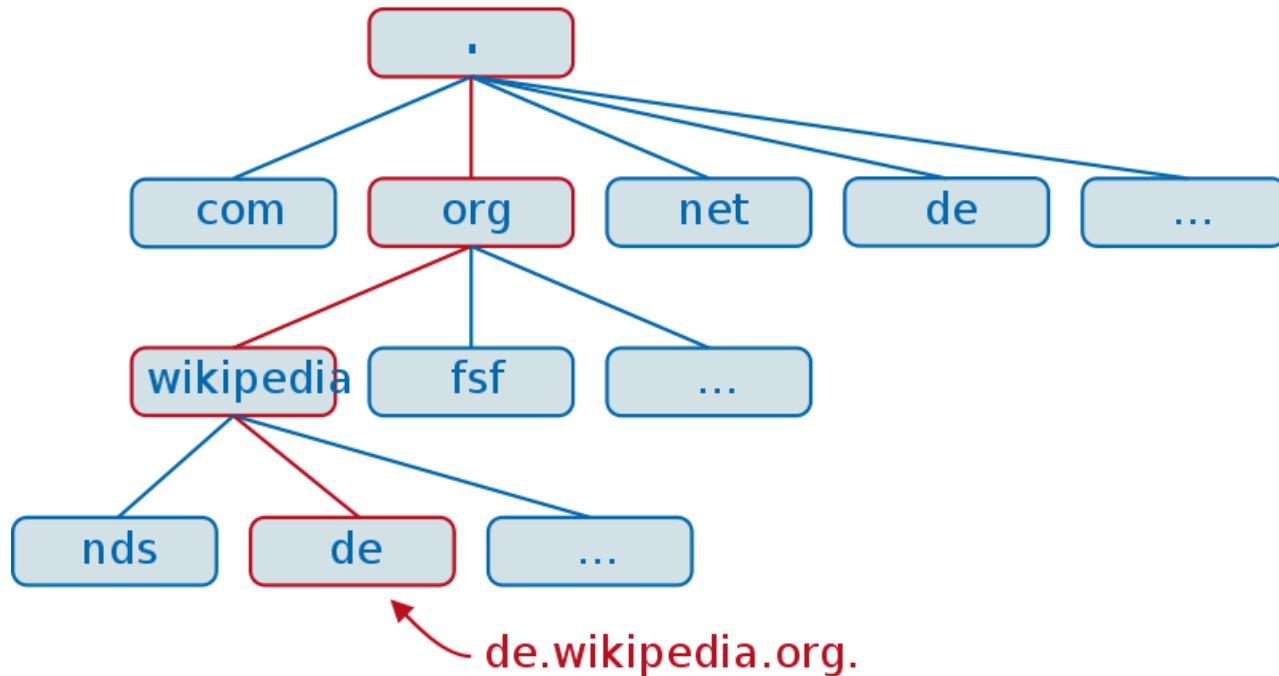
# IP-ADRESSE

- Host-Angabe in URL: IP-Adresse oder Domain-Name
- IPv4: 32 Bit, IPv6: 128 Bit
- Beispiele:

173.194.40.95

243f:6a88:85a3:08d3:1319:8a2e:0370:7344

# DOMAIN NAME SYSTEM (DNS)



IP-Adressen schwer  
zu merken, Abhilfe:  
Domain-Namen

# PROTOKOLL: HTTP

- Einfaches, textbasiertes Protokoll
- Beispiel: Anfrage an Server

```
GET /~bkrt/hallo.html HTTP/1.1
```

Host: dublin.zhaw.ch

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X...) Gecko/20100101 Firefox/22.0

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

Accept-Language: de-de,de;q=0.8,en-us;q=0.5,en;q=0.3

Accept-Encoding: gzip, deflate

Connection: keep-alive

## Speaker notes

### Antwort vom Server

```
HTTP/1.1 200 OK
Date: Mon, 15 Jul 2013 17:10:56 GMT
Server: Apache/2.2.15 (CentOS)
Last-Modified: Wed, 17 Oct 2012 08:10:22 GMT
ETag: "5b018a-af-4cc3ccd575780"
Accept-Ranges: bytes
Content-Length: 175
Connection: close
Content-Type: text/html; charset=UTF-8
```

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Hallo</title>
  </head>
  <body>
    <h1>Hallo</h1>
    <p>Ich bin eine Webseite</p>
  </body>
</html>
```

Hypertext Transfer Protocol - HTTP/1.1 (RFC 2616)

<https://www.ietf.org/rfc/rfc2616.txt>

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

# DIE SPRACHEN DES WEB

- HTML und CSS bereits bekannt  
(Vorkenntnisse, Vorkurs)
- Daher hier nur eine schnelle Zusammenfassung...

# ÜBERSICHT

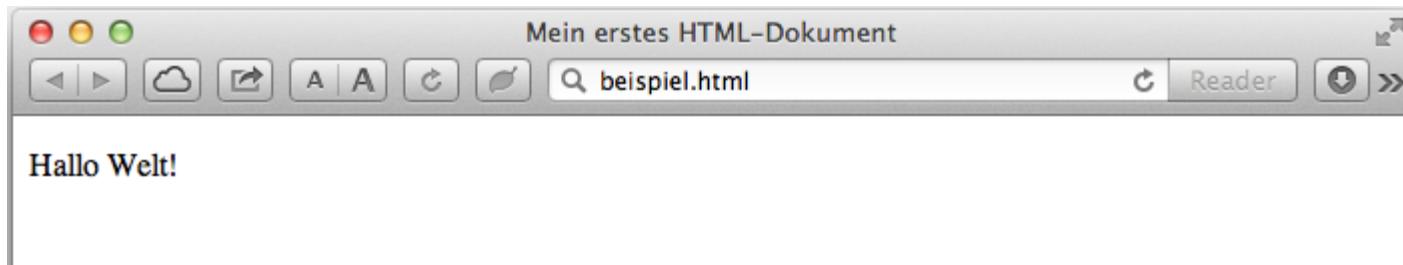
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
  - Markup mit HTML
  - Darstellung mit CSS
  - Verhalten mit JavaScript
- Web-Standards und APIs

# MARKUP MIT HTML

- **Markup Language:** Tags (`<p>`)
- **Hypertext:** Links
- Multimedia: eingebettete Multimedia-Elemente
- Gestaltung: verbundene CSS-Datei(en)
- Verhalten: verbundene JavaScript-Datei(en)
- Verschiedene Content-Types: `text/html`, `image/jpeg`, ...

# MARKUP MIT HTML

```
1 <!DOCTYPE html>
2 <html lang="de">
3   <head>
4     <title>Mein erstes HTML-Dokument</title>
5     <meta charset="utf-8" />
6   </head>
7
8   <body>
9     <p>Hallo Welt!</p>
10  </body>
11 </html>
```



# MARKUP MIT HTML

- Es geht um **Inhalt** und **Struktur** von Dokumenten
- Was bedeuten die einzelnen Teile (Überschrift? Absatz?)
- Darstellung und Verhalten explizit ausgeklammert
- Ohne CSS-Stylesheet: Standard-Darstellung im Browser

## Speaker notes

Eine grosse Diskussion derzeit unter Webentwicklern ist, ob es eine gute Idee ist, Tools wie **Tailwind CSS** einzusetzen. Kurz: Tailwind CSS stellt eine grosse Zahl von CSS-Klassen zur Verfügung, um bestimmte Darstellungen zu erreichen, z.B.:

```
<figure class="md:flex bg-slate-100 rounded-xl p-8 md:p-0 dark:bg-slate-800">
  ...
</figure>
```

Damit nicht bei jedem Laden die gesamte Liste der CSS-Regeln geladen werden muss, kann die CSS-Datei in einem Build-Schritt auf die tatsächlich verwendeten Klassen reduziert werden.

Viele Entwickler sehen diesen Ansatz als Rückschritt um ein paar Jahrzehnte. Damals hat man Darstellung und Inhalt/Struktur ebenfalls vermischt – es gab Attribute wie font – oder man hat die Styles inline in den HTML-Code übernommen:

```
<figure style="display:flex, border-radius:5px, ...">
  ...
</figure>
```

Im Gegensatz zu diesem Beispiel bezieht Tailwind die Display-Grösse mit ein, was sicher ein Vorteil ist. Eigentlich sollte es aber beispielsweise so sein:

```
<figure class="employee-image">
  ...
</figure>
```

Damit hat man den Inhalt beschrieben, die Darstellung sollte an anderer, zentraler Stelle beschrieben werden. Vorteile des Tailwind-Ansatzes: man hat schnell eine Seite mit guter Darstellung zusammengeschustert, muss sich kaum mit CSS auseinandersetzen, und kann Code, der sich bewährt hat, einfach per Copy&Paste übertragen. Nachteil dieses Ansatzes: Trennung von Inhalt und Darstellung gehen verloren (wie wenn man in Word die Darstellungsoptionen für jeden Textbereich angibt statt Styles wie Titel, Text oder Liste zu verwenden) und man hat unnötig viel Redundanz im Code.

Tailwind, and the death of web craftsmanship

<https://pdx.su/blog/2023-07-26-tailwind-and-the-death-of-craftsmanship/>

# HTML: NOCH EIN BEISPIEL

```
1 <!DOCTYPE html>
2 <html lang="de">
3   <head> ... </head>
4   <body>
5     <section>
6       <header>
7         
8       </header>
9       <article>
10        <h1>Webseiten</h1>
11        <p>Die Grundidee ist ziemlich einfach:</p>
12        <p>Das <strong>World Wide Web</strong> (heute meist kurz: das Web)
13          basiert auf einer <em>Client-Server-Architektur</em>. Der Client
14          ist meist ein grafischer <em>Browser</em>. ...</p>
15        </article>
16      </section>
17    </body>
18 </html>
```

## Speaker notes

Beispiel komplett:

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <title>Mein zweites HTML-Dokument</title>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="styles/screen.css">
  </head>

  <body>
    <section>
      <header>
        
      </header>

      <article>
        <h1>Webseiten</h1>

        <p>Die Grundidee ist ziemlich einfach:</p>

        <p>Das <strong>World Wide Web</strong> (heute meist kurz: das Web)
          basiert auf einer <em>Client-Server-Architektur</em>. Der Client
          ist meist ein grafischer <em>Browser</em>. ....</p>
      </article>
    </section>
  </body>
</html>
```

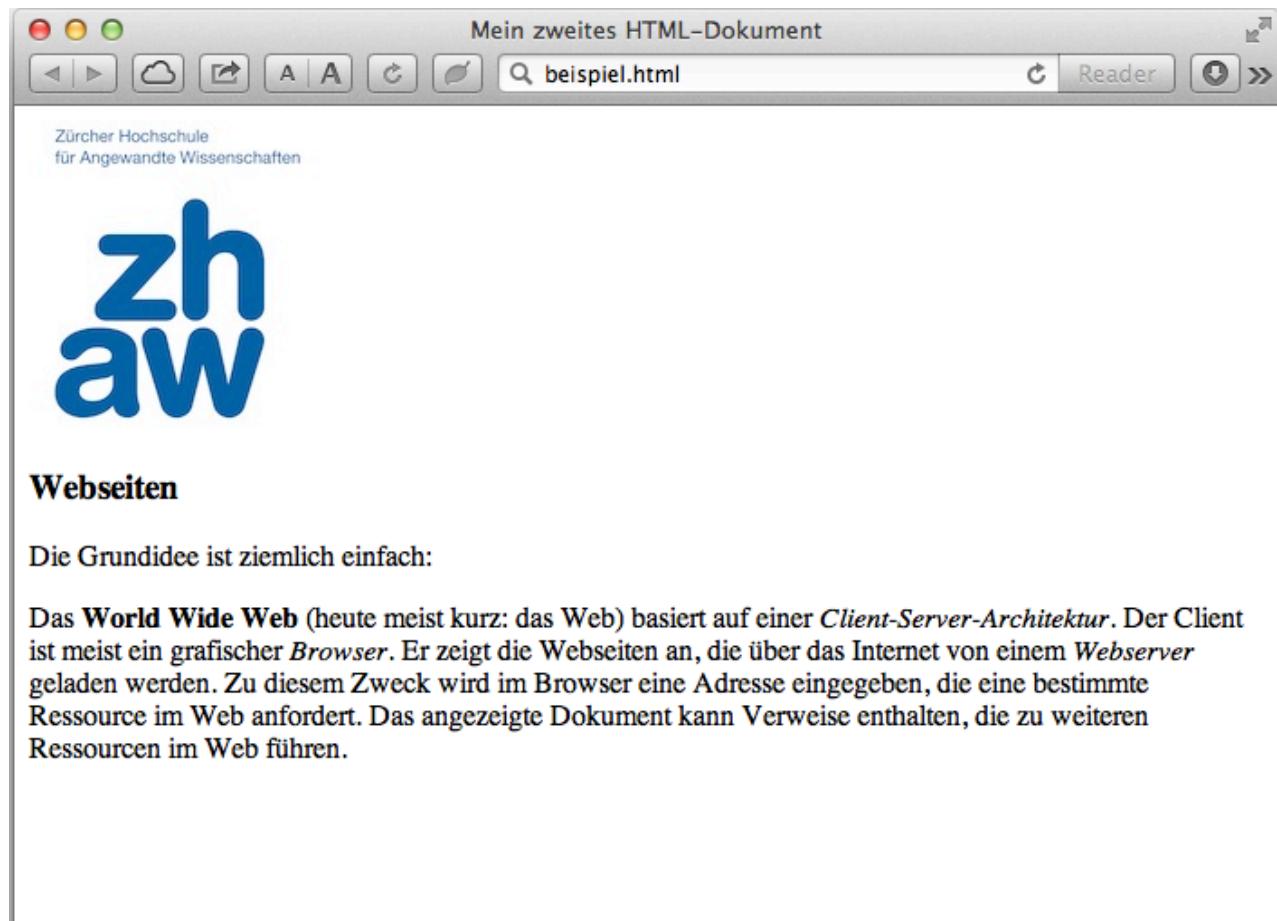
HTML ist eine Auszeichnungssprache (Markup Language). Für die Auszeichnung (das *Markup*) der Inhalte werden *Tags* verwendet, etwa <h1>...</h1> für Überschriften oder <p>...</p> für Absätze. Die meisten HTML-Elemente haben ein *Start-Tag* und ein *End-Tag* und umklammern damit den Inhalt, der weitere Tags enthalten kann. Dadurch entsteht eine hierarchische Struktur der Dokuments. *Leere Elemente* haben keinen Inhalt, zum Beispiel <meta ...>, <img ...>.

Das *Start-Tag* (oder im Fall eines leeren Elements das *Tag*) kann auch Attribute haben, welche das Element näher beschreiben. Die korrekte Schreibweise ist der Name des Attributs, ein =-Zeichen ohne Leerzeichen davor oder danach, und schliesslich der Attributwert in Anführungszeichen. Nach einem Leerzeichen können weitere Attribute folgen.

```

```

# STANDARD-DARSTELLUNG (OHNE CSS)

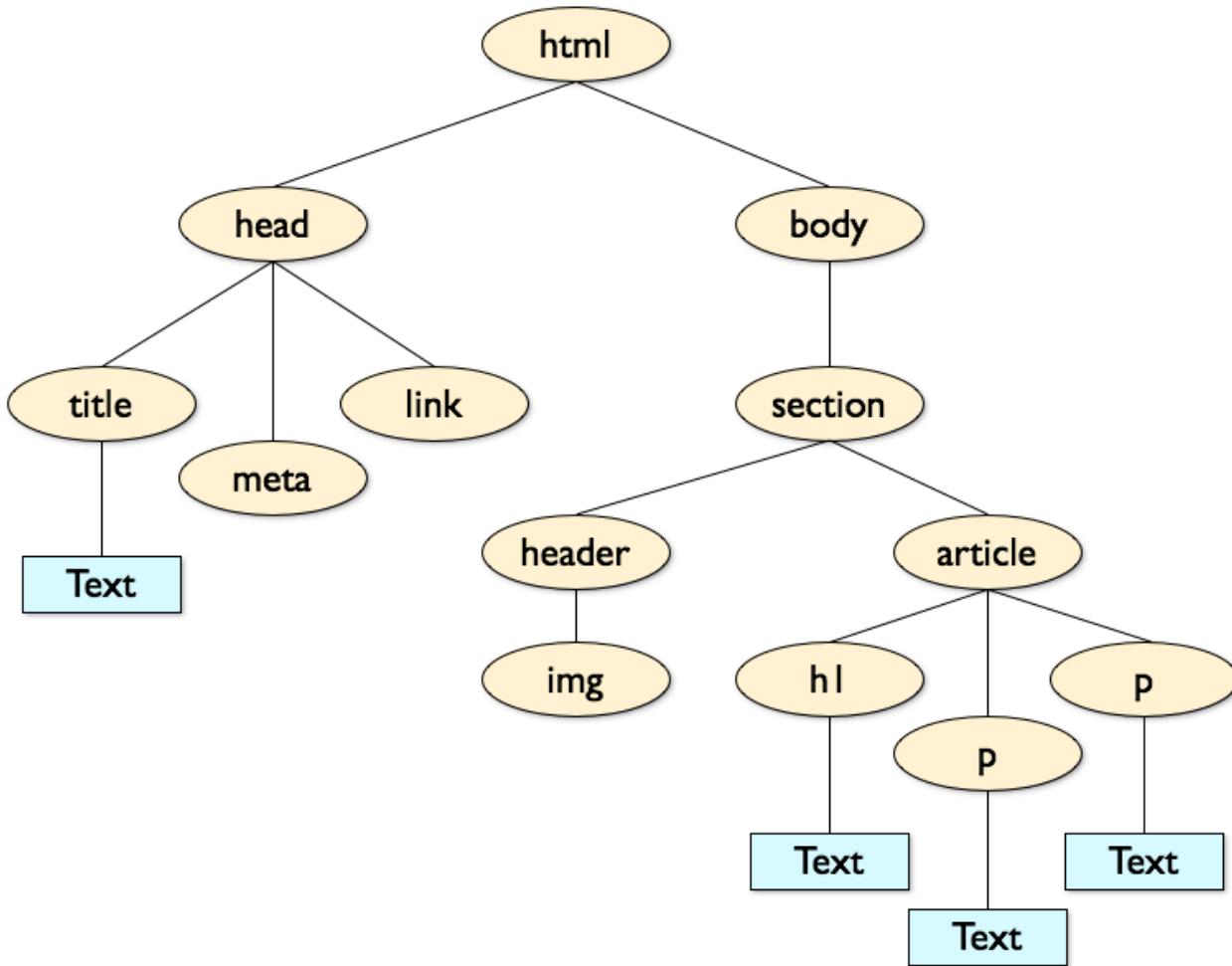


## Webseiten

Die Grundidee ist ziemlich einfach:

Das **World Wide Web** (heute meist kurz: das Web) basiert auf einer *Client-Server-Architektur*. Der Client ist meist ein grafischer *Browser*. Er zeigt die Webseiten an, die über das Internet von einem *Webserver* geladen werden. Zu diesem Zweck wird im Browser eine Adresse eingegeben, die eine bestimmte Ressource im Web anfordert. Das angezeigte Dokument kann Verweise enthalten, die zu weiteren Ressourcen im Web führen.

# HIERARCHISCHE STRUKTUR



## Speaker notes

Bestimmte Entwickler-Tools im Browser können diese Hierarchie auch explizit im Browser hervorheben, wie das folgende Bild zeigt.

```
<!DOCTYPE html>
<HTML lang="de">

<head>
<title>
Mein zweites HTML-Dokument
</title>
<meta charset="utf-8">
<link rel="stylesheet" href="styles/screen.css">
</head>

<body>
<section>
<header>

</header>

<article>
<h1>
Webseiten
</h1>

<p>
Die Grundidee ist ziemlich einfach:
</p>

<p>
...
</p>
</article>
</section>
</body>
</HTML>
```

# ÜBERSICHT

- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
  - Markup mit HTML
  - Darstellung mit CSS
  - Verhalten mit JavaScript
- Web-Standards und APIs

# CASCADING STYLE SHEETS (CSS)

```
1 @charset "utf-8";
2
3 body {
4     background-color: #999;
5 }
6 article {
7     margin-left: 200px;
8     font-family: 'Georgia Pro', georgia, serif;
9     line-height: 140%;
10 }
11 p {
12     font-size: 3em;
13     color: blue;
14 }
15 ...
```

# DARSTELLUNG OHNE/MIT CSS

The screenshot displays a web browser window with two tabs. The left tab, titled "Mein zweites", shows the Zürcher Hochschule für Angewandte Wissenschaften (ZHAW) website. It features the blue "zhaw" logo and a section titled "Webseiten" with the following text:

Die Grundidee ist ziemlich einfach:

Das **World Wide Web** (heute meist kurz: das Web) ist meist ein grafischer *Browser*. Er zeigt die Webseiten an, die über das Internet von einem *Webserver* geladen werden. Zu diesem Zweck wird im Browser eine Adresse eingegeben, die eine bestimmte Ressource im Web anfordert. Das angezeigte Dokument kann Verweise enthalten, die zu weiteren Ressourcen im Web führen.

The right tab, also titled "Mein zweites HTML-Dokument", shows the same content but with more detailed styling. The "zhaw" logo is larger and more prominent. The heading "Webseiten" is bolded. The explanatory text is also more detailed, using **World Wide Web** and *Client-Server-Architektur* in italics. The overall layout is cleaner and more professional due to the applied CSS.

# ÜBERSICHT

- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
  - Markup mit HTML
  - Darstellung mit CSS
  - **Verhalten mit JavaScript**
- Web-Standards und APIs

# VERHALTEN MIT JAVASCRIPT

```
1 let pElem = document.querySelector('p')
2 pElem.addEventListener('click', function () {
3     let newText = document.createTextNode("removed!")
4     pElem.replaceChild(newText, pElem.childNodes[0])
5 })
```

- Klick auf den Text ersetzt ihn durch "removed!"
- Element des Dokuments auswählen
- Reaktion auf Ereignis festlegen

# JAVASCRIPT

- Veröffentlich 1995 in Vorversion des Netscape Navigators 2.0
- Unter Zeitdruck entwickelt von Brendan Eich
- Ziel: Scripts um Webseiten dynamischer zu machen
- Zunächst: **LiveScript**, dann **JavaScript** (Marketing)
- **JavaScript und Java haben wenig gemeinsam (!)**

# JAVASCRIPT HEUTE

- Web-Apps mit vielen tausend Codezeilen (z.B. Maps)
- Einsatz in nativen Applikationen (z.B. VSCode)
- Performant: Just-in-time Compiler (JIT)
- Zielsprache für andere Sprachen (z.B. TypeScript)

JavaScript ist heute eine der wichtigsten und meistverwendeten Programmiersprachen

# JAVASCRIPT PLATTFORMEN

- **Web-Browser** mit verschiedenen JavaScript-Engines und diversen APIs für die Browser-Plattform (DOM, Storage, ...)
- **Node.js** als browserunabhängige JavaScript-Plattform, ebenfalls mit speziellen APIs (Filesystem, Web-Server, ...)
- Alternativen zu Node.js wie **Deno** oder **Bun**

# JAVASCRIPT-ALTERNATIVEN

- Sprachen, welche nach JavaScript übersetzt werden, u.a.
  - [TypeScript](#)
  - [ReScript](#)
  - [ClojureScript](#)
  - neueres JavaScript zu älterem JavaScript
- Transpiler (z.B. [Babel](#))

# WEB-ASSEMBLY

- Neuere Entwicklung
- Virtuelle Maschine
- Diverse Sprachen zu Web-Assembly kompilierbar

## Speaker notes

Damit auch möglich, Sprachen wie Rust oder Kotlin im Web einzusetzen.

Dabei soll aber nicht verschwiegen werden, dass JavaScript und das Web gemeinsam gewachsen und somit eng verzahnt sind. Das gilt vor allem für das DOM, das Document Object Model. DOM-Manipulationen wird man auch in Zukunft eher in JavaScript oder einer eng verwandten Sprache machen.

# ÜBERSICHT

- Worum geht's in WBE?
- Organisatorisches
- Internet und WWW
- Client-Server-Architektur
- Die Sprachen des Web: HTML, CSS, JavaScript
- Web-Standards und APIs

# WORLD WIDE WEB CONSORTIUM

- Kurz: W3C
- Standardisiert Technologien des World Wide Web
- Gegründet 1994 am MIT
- Gründer und Vorsitzender: Tim Berners-Lee

Leider schwindet in letzter Zeit der Einfluss des W3C auf die weitere Entwicklung des Web, da die Browser-Hersteller (dominanter Player: Google) mehr und mehr die Richtung vorgeben

# WHATWG

- Web Hypertext Application Technology Working Group
- Gründer: Apple, Mozilla, Opera, später: Microsoft, Google
- Grund: Unzufriedenheit mit W3C-Ausrichtung
- Eigene Standards der WHATWG
- Zeitweise parallel: HTML des W3C, HTML der WHATWG
- Heute: **HTML Living Standard** der WHATWG

<https://html.spec.whatwg.org/multipage/>

# TECHNOLOGIEN

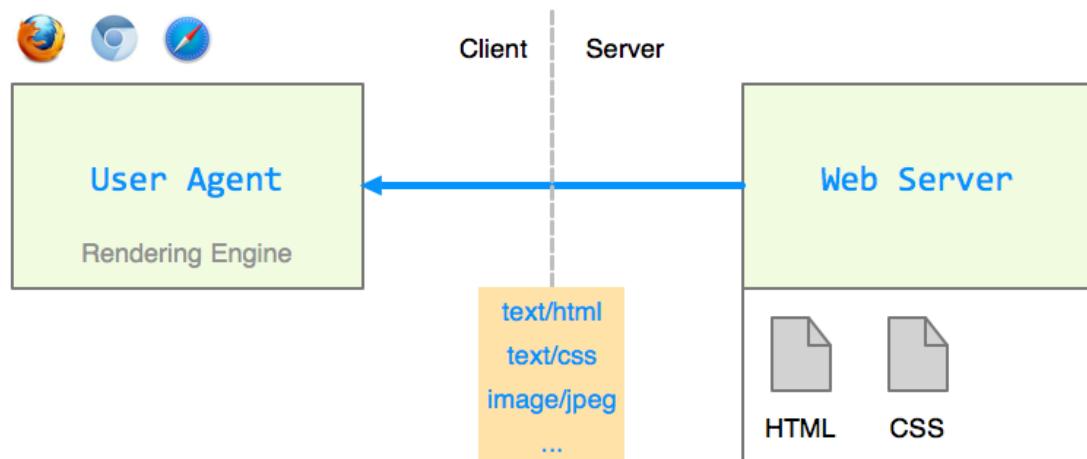
## Client-seitig

- Beschränkt auf das, was der Browser kann
- HTML + CSS + JavaScript + noch ein paar Sachen
- → Front-end Entwickler

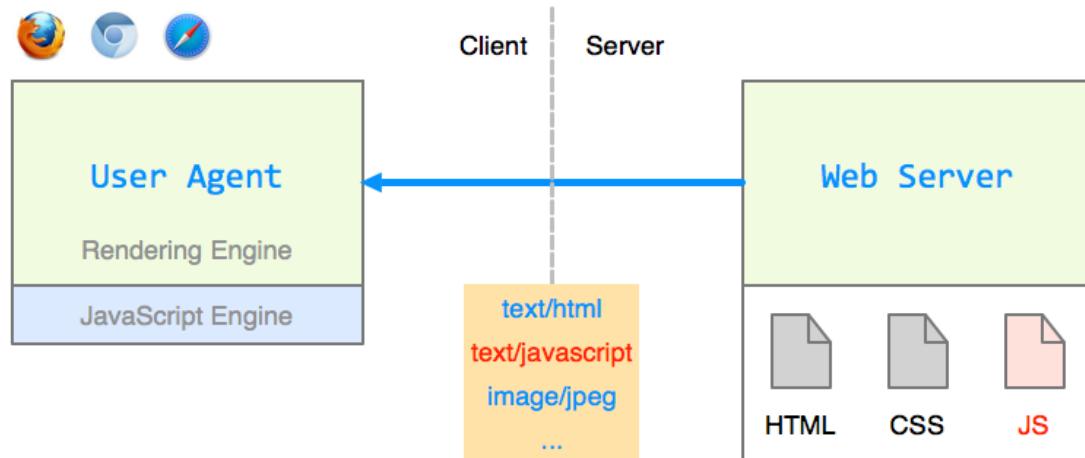
## Server-seitig

- Praktisch unbeschränkt: Plattform, Programmiersprache, ...
- Erzeugt und gesendet wird das, was der Browser kann
- → Back-end Entwickler

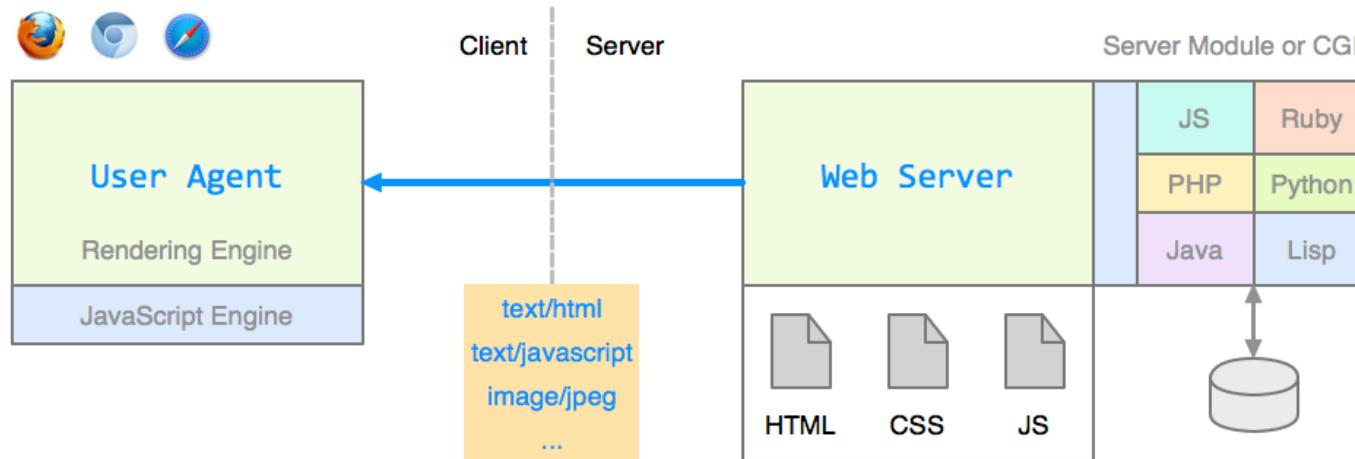
# STATISCHE WEBSEITEN



# CLIENTSEITIGE PROGRAMMLOGIK



# SERVERSEITIGE PROGRAMMLOGIK



# ENTWICKLUNG

- Statische Webseiten
- Generierte Inhalte (CGI z.B. Shell Scripts, Perl)
- Serverseitig eingebettete Scriptsprachen (PHP)
- Client Scripting und Applets (JavaScript, Java Applets, Flash)
- Enterprise Application Server (Java, Java EE)
- MVC Serveranwendungen (Rails, Django)
- JavaScript serverseitig

# UMFANGREICHES GEBIET

The screenshot shows the Can I Use website interface. At the top, there's a navigation bar with links for Home, News, and a date (June 26, 2020). To the right of the navigation is a "Compare browsers" button. The main content area has three columns: "CSS", "HTML5", and "SVG". Each column contains a list of features with a small square icon indicating support. Below these columns is a "JS API" section with a similar list. On the far right, there's a sidebar titled "Can I Use" with sections for CSS, HTML5, and SVG, each listing many specific features and their browser support.

## CSS

- ::first-letter CSS pseudo-element selector
- ::placeholder CSS pseudo-element
- ::selection CSS pseudo-element
- :dir() CSS pseudo-class
- :has() CSS relational pseudo-class
- :in-range and :out-of-range CSS pseudo-classes
- :is() CSS pseudo-class
- :placeholder-shown CSS pseudo-class
- @font-face Web fonts
- Blending of HTML/SVG elements
- calc() as CSS unit value
- Case-insensitive CSS attribute selectors
- ch (character) unit
- CSS 2.1 selectors
- CSS ::marker pseudo-element
- CSS :any-link selector

## HTML5

- accept attribute for file input
- Add to home screen (A2HS)
- async attribute for external scripts
- Attributes for form submission
- Audio element
- Audio Tracks
- Autofocus attribute
- Canvas (basic support)
- Canvas blend modes
- classList (DOMTokenList)
- Color input type
- contenteditable attribute (basic support)
- Custom Elements (V1)
- Custom protocol handling
- Datalist element
- dataset & data-\* attributes
- Date and time input types

## SVG

- SVG (basic support)
- SVG effects for HTML
- SVG filters
- SVG fonts
- SVG fragment identifiers
- SVG in HTML img element
- SVG SMIL animation
- **All SVG features**

## JS API

- AbortController & AbortSignal
- Accelerometer
- Ambient Light Sensor
- Auxclick
- Base64 encoding and decoding
- Basic console logging functions

Can I Use	?	Settings
<ul style="list-style-type: none"><li>■ CSS 1.0</li><li>■ CSS 2.0</li><li>■ CSS 2.1</li><li>■ CSS 3.0</li><li>■ CSS 4.0</li><li>■ CSS 5.0</li><li>■ CSS 6.0</li><li>■ CSS 7.0</li><li>■ CSS 8.0</li><li>■ CSS 9.0</li><li>■ CSS 10.0</li><li>■ CSS 11.0</li><li>■ CSS 12.0</li><li>■ CSS 13.0</li><li>■ CSS 14.0</li><li>■ CSS 15.0</li><li>■ CSS 16.0</li><li>■ CSS 17.0</li><li>■ CSS 18.0</li><li>■ CSS 19.0</li><li>■ CSS 20.0</li><li>■ CSS 21.0</li><li>■ CSS 22.0</li><li>■ CSS 23.0</li><li>■ CSS 24.0</li><li>■ CSS 25.0</li><li>■ CSS 26.0</li><li>■ CSS 27.0</li><li>■ CSS 28.0</li><li>■ CSS 29.0</li><li>■ CSS 30.0</li><li>■ CSS 31.0</li><li>■ CSS 32.0</li><li>■ CSS 33.0</li><li>■ CSS 34.0</li><li>■ CSS 35.0</li><li>■ CSS 36.0</li><li>■ CSS 37.0</li><li>■ CSS 38.0</li><li>■ CSS 39.0</li><li>■ CSS 40.0</li><li>■ CSS 41.0</li><li>■ CSS 42.0</li><li>■ CSS 43.0</li><li>■ CSS 44.0</li><li>■ CSS 45.0</li><li>■ CSS 46.0</li><li>■ CSS 47.0</li><li>■ CSS 48.0</li><li>■ CSS 49.0</li><li>■ CSS 50.0</li><li>■ CSS 51.0</li><li>■ CSS 52.0</li><li>■ CSS 53.0</li><li>■ CSS 54.0</li><li>■ CSS 55.0</li><li>■ CSS 56.0</li><li>■ CSS 57.0</li><li>■ CSS 58.0</li><li>■ CSS 59.0</li><li>■ CSS 60.0</li><li>■ CSS 61.0</li><li>■ CSS 62.0</li><li>■ CSS 63.0</li><li>■ CSS 64.0</li><li>■ CSS 65.0</li><li>■ CSS 66.0</li><li>■ CSS 67.0</li><li>■ CSS 68.0</li><li>■ CSS 69.0</li><li>■ CSS 70.0</li><li>■ CSS 71.0</li><li>■ CSS 72.0</li><li>■ CSS 73.0</li><li>■ CSS 74.0</li><li>■ CSS 75.0</li><li>■ CSS 76.0</li><li>■ CSS 77.0</li><li>■ CSS 78.0</li><li>■ CSS 79.0</li><li>■ CSS 80.0</li><li>■ CSS 81.0</li><li>■ CSS 82.0</li><li>■ CSS 83.0</li><li>■ CSS 84.0</li><li>■ CSS 85.0</li><li>■ CSS 86.0</li><li>■ CSS 87.0</li><li>■ CSS 88.0</li><li>■ CSS 89.0</li><li>■ CSS 90.0</li><li>■ CSS 91.0</li><li>■ CSS 92.0</li><li>■ CSS 93.0</li><li>■ CSS 94.0</li><li>■ CSS 95.0</li><li>■ CSS 96.0</li><li>■ CSS 97.0</li><li>■ CSS 98.0</li><li>■ CSS 99.0</li><li>■ CSS 100.0</li><li>■ CSS 101.0</li><li>■ CSS 102.0</li><li>■ CSS 103.0</li><li>■ CSS 104.0</li><li>■ CSS 105.0</li><li>■ CSS 106.0</li><li>■ CSS 107.0</li><li>■ CSS 108.0</li><li>■ CSS 109.0</li><li>■ CSS 110.0</li><li>■ CSS 111.0</li><li>■ CSS 112.0</li><li>■ CSS 113.0</li><li>■ CSS 114.0</li><li>■ CSS 115.0</li><li>■ CSS 116.0</li><li>■ CSS 117.0</li><li>■ CSS 118.0</li><li>■ CSS 119.0</li><li>■ CSS 120.0</li><li>■ CSS 121.0</li><li>■ CSS 122.0</li><li>■ CSS 123.0</li><li>■ CSS 124.0</li><li>■ CSS 125.0</li><li>■ CSS 126.0</li><li>■ CSS 127.0</li><li>■ CSS 128.0</li><li>■ CSS 129.0</li><li>■ CSS 130.0</li><li>■ CSS 131.0</li><li>■ CSS 132.0</li><li>■ CSS 133.0</li><li>■ CSS 134.0</li><li>■ CSS 135.0</li><li>■ CSS 136.0</li><li>■ CSS 137.0</li><li>■ CSS 138.0</li><li>■ CSS 139.0</li><li>■ CSS 140.0</li><li>■ CSS 141.0</li><li>■ CSS 142.0</li><li>■ CSS 143.0</li><li>■ CSS 144.0</li><li>■ CSS 145.0</li><li>■ CSS 146.0</li><li>■ CSS 147.0</li><li>■ CSS 148.0</li><li>■ CSS 149.0</li><li>■ CSS 150.0</li><li>■ CSS 151.0</li><li>■ CSS 152.0</li><li>■ CSS 153.0</li><li>■ CSS 154.0</li><li>■ CSS 155.0</li><li>■ CSS 156.0</li><li>■ CSS 157.0</li><li>■ CSS 158.0</li><li>■ CSS 159.0</li><li>■ CSS 160.0</li><li>■ CSS 161.0</li><li>■ CSS 162.0</li><li>■ CSS 163.0</li><li>■ CSS 164.0</li><li>■ CSS 165.0</li><li>■ CSS 166.0</li><li>■ CSS 167.0</li><li>■ CSS 168.0</li><li>■ CSS 169.0</li><li>■ CSS 170.0</li><li>■ CSS 171.0</li><li>■ CSS 172.0</li><li>■ CSS 173.0</li><li>■ CSS 174.0</li><li>■ CSS 175.0</li><li>■ CSS 176.0</li><li>■ CSS 177.0</li><li>■ CSS 178.0</li><li>■ CSS 179.0</li><li>■ CSS 180.0</li><li>■ CSS 181.0</li><li>■ CSS 182.0</li><li>■ CSS 183.0</li><li>■ CSS 184.0</li><li>■ CSS 185.0</li><li>■ CSS 186.0</li><li>■ CSS 187.0</li><li>■ CSS 188.0</li><li>■ CSS 189.0</li><li>■ CSS 190.0</li><li>■ CSS 191.0</li><li>■ CSS 192.0</li><li>■ CSS 193.0</li><li>■ CSS 194.0</li><li>■ CSS 195.0</li><li>■ CSS 196.0</li><li>■ CSS 197.0</li><li>■ CSS 198.0</li><li>■ CSS 199.0</li><li>■ CSS 200.0</li><li>■ CSS 201.0</li><li>■ CSS 202.0</li><li>■ CSS 203.0</li><li>■ CSS 204.0</li><li>■ CSS 205.0</li><li>■ CSS 206.0</li><li>■ CSS 207.0</li><li>■ CSS 208.0</li><li>■ CSS 209.0</li><li>■ CSS 210.0</li><li>■ CSS 211.0</li><li>■ CSS 212.0</li><li>■ CSS 213.0</li><li>■ CSS 214.0</li><li>■ CSS 215.0</li><li>■ CSS 216.0</li><li>■ CSS 217.0</li><li>■ CSS 218.0</li><li>■ CSS 219.0</li><li>■ CSS 220.0</li><li>■ CSS 221.0</li><li>■ CSS 222.0</li><li>■ CSS 223.0</li><li>■ CSS 224.0</li><li>■ CSS 225.0</li><li>■ CSS 226.0</li><li>■ CSS 227.0</li><li>■ CSS 228.0</li><li>■ CSS 229.0</li><li>■ CSS 230.0</li><li>■ CSS 231.0</li><li>■ CSS 232.0</li><li>■ CSS 233.0</li><li>■ CSS 234.0</li><li>■ CSS 235.0</li><li>■ CSS 236.0</li><li>■ CSS 237.0</li><li>■ CSS 238.0</li><li>■ CSS 239.0</li><li>■ CSS 240.0</li><li>■ CSS 241.0</li><li>■ CSS 242.0</li><li>■ CSS 243.0</li><li>■ CSS 244.0</li><li>■ CSS 245.0</li><li>■ CSS 246.0</li><li>■ CSS 247.0</li><li>■ CSS 248.0</li><li>■ CSS 249.0</li><li>■ CSS 250.0</li><li>■ CSS 251.0</li><li>■ CSS 252.0</li><li>■ CSS 253.0</li><li>■ CSS 254.0</li><li>■ CSS 255.0</li><li>■ CSS 256.0</li><li>■ CSS 257.0</li><li>■ CSS 258.0</li><li>■ CSS 259.0</li><li>■ CSS 260.0</li><li>■ CSS 261.0</li><li>■ CSS 262.0</li><li>■ CSS 263.0</li><li>■ CSS 264.0</li><li>■ CSS 265.0</li><li>■ CSS 266.0</li><li>■ CSS 267.0</li><li>■ CSS 268.0</li><li>■ CSS 269.0</li><li>■ CSS 270.0</li><li>■ CSS 271.0</li><li>■ CSS 272.0</li><li>■ CSS 273.0</li><li>■ CSS 274.0</li><li>■ CSS 275.0</li><li>■ CSS 276.0</li><li>■ CSS 277.0</li><li>■ CSS 278.0</li><li>■ CSS 279.0</li><li>■ CSS 280.0</li><li>■ CSS 281.0</li><li>■ CSS 282.0</li><li>■ CSS 283.0</li><li>■ CSS 284.0</li><li>■ CSS 285.0</li><li>■ CSS 286.0</li><li>■ CSS 287.0</li><li>■ CSS 288.0</li><li>■ CSS 289.0</li><li>■ CSS 290.0</li><li>■ CSS 291.0</li><li>■ CSS 292.0</li><li>■ CSS 293.0</li><li>■ CSS 294.0</li><li>■ CSS 295.0</li><li>■ CSS 296.0</li><li>■ CSS 297.0</li><li>■ CSS 298.0</li><li>■ CSS 299.0</li><li>■ CSS 300.0</li><li>■ CSS 301.0</li><li>■ CSS 302.0</li><li>■ CSS 303.0</li><li>■ CSS 304.0</li><li>■ CSS 305.0</li><li>■ CSS 306.0</li><li>■ CSS 307.0</li><li>■ CSS 308.0</li><li>■ CSS 309.0</li><li>■ CSS 310.0</li><li>■ CSS 311.0</li><li>■ CSS 312.0</li><li>■ CSS 313.0</li><li>■ CSS 314.0</li><li>■ CSS 315.0</li><li>■ CSS 316.0</li><li>■ CSS 317.0</li><li>■ CSS 318.0</li><li>■ CSS 319.0</li><li>■ CSS 320.0</li><li>■ CSS 321.0</li><li>■ CSS 322.0</li><li>■ CSS 323.0</li><li>■ CSS 324.0</li><li>■ CSS 325.0</li><li>■ CSS 326.0</li><li>■ CSS 327.0</li><li>■ CSS 328.0</li><li>■ CSS 329.0</li><li>■ CSS 330.0</li><li>■ CSS 331.0</li><li>■ CSS 332.0</li><li>■ CSS 333.0</li><li>■ CSS 334.0</li><li>■ CSS 335.0</li><li>■ CSS 336.0</li><li>■ CSS 337.0</li><li>■ CSS 338.0</li><li>■ CSS 339.0</li><li>■ CSS 340.0</li><li>■ CSS 341.0</li><li>■ CSS 342.0</li><li>■ CSS 343.0</li><li>■ CSS 344.0</li><li>■ CSS 345.0</li><li>■ CSS 346.0</li><li>■ CSS 347.0</li><li>■ CSS 348.0</li><li>■ CSS 349.0</li><li>■ CSS 350.0</li><li>■ CSS 351.0</li><li>■ CSS 352.0</li><li>■ CSS 353.0</li><li>■ CSS 354.0</li><li>■ CSS 355.0</li><li>■ CSS 356.0</li><li>■ CSS 357.0</li><li>■ CSS 358.0</li><li>■ CSS 359.0</li><li>■ CSS 360.0</li><li>■ CSS 361.0</li><li>■ CSS 362.0</li><li>■ CSS 363.0</li><li>■ CSS 364.0</li><li>■ CSS 365.0</li><li>■ CSS 366.0</li><li>■ CSS 367.0</li><li>■ CSS 368.0</li><li>■ CSS 369.0</li><li>■ CSS 370.0</li><li>■ CSS 371.0</li><li>■ CSS 372.0</li><li>■ CSS 373.0</li><li>■ CSS 374.0</li><li>■ CSS 375.0</li><li>■ CSS 376.0</li><li>■ CSS 377.0</li><li>■ CSS 378.0</li><li>■ CSS 379.0</li><li>■ CSS 380.0</li><li>■ CSS 381.0</li><li>■ CSS 382.0</li><li>■ CSS 383.0</li><li>■ CSS 384.0</li><li>■ CSS 385.0</li><li>■ CSS 386.0</li><li>■ CSS 387.0</li><li>■ CSS 388.0</li><li>■ CSS 389.0</li><li>■ CSS 390.0</li><li>■ CSS 391.0</li><li>■ CSS 392.0</li><li>■ CSS 393.0</li><li>■ CSS 394.0</li><li>■ CSS 395.0</li><li>■ CSS 396.0</li><li>■ CSS 397.0</li><li>■ CSS 398.0</li><li>■ CSS 399.0</li><li>■ CSS 400.0</li><li>■ CSS 401.0</li><li>■ CSS 402.0</li><li>■ CSS 403.0</li><li>■ CSS 404.0</li><li>■ CSS 405.0</li><li>■ CSS 406.0</li><li>■ CSS 407.0</li><li>■ CSS 408.0</li><li>■ CSS 409.0</li><li>■ CSS 410.0</li><li>■ CSS 411.0</li><li>■ CSS 412.0</li><li>■ CSS 413.0</li><li>■ CSS 414.0</li><li>■ CSS 415.0</li><li>■ CSS 416.0</li><li>■ CSS 417.0</li><li>■ CSS 418.0</li><li>■ CSS 419.0</li><li>■ CSS 420.0</li><li>■ CSS 421.0</li><li>■ CSS 422.0</li><li>■ CSS 423.0</li><li>■ CSS 424.0</li><li>■ CSS 425.0</li><li>■ CSS 426.0</li><li>■ CSS 427.0</li><li>■ CSS 428.0</li><li>■ CSS 429.0</li><li>■ CSS 430.0</li><li>■ CSS 431.0</li><li>■ CSS 432.0</li><li>■ CSS 433.0</li><li>■ CSS 434.0</li><li>■ CSS 435.0</li><li>■ CSS 436.0</li><li>■ CSS 437.0</li><li>■ CSS 438.0</li><li>■ CSS 439.0</li><li>■ CSS 440.0</li><li>■ CSS 441.0</li><li>■ CSS 442.0</li><li>■ CSS 443.0</li><li>■ CSS 444.0</li><li>■ CSS 445.0</li><li>■ CSS 446.0</li><li>■ CSS 447.0</li><li>■ CSS 448.0</li><li>■ CSS 449.0</li><li>■ CSS 450.0</li><li>■ CSS 451.0</li><li>■ CSS 452.0</li><li>■ CSS 453.0</li><li>■ CSS 454.0</li><li>■ CSS 455.0</li><li>■ CSS 456.0</li><li>■ CSS 457.0</li><li>■ CSS 458.0</li><li>■ CSS 459.0</li><li>■ CSS 460.0</li><li>■ CSS 461.0</li><li>■ CSS 462.0</li><li>■ CSS 463.0</li><li>■ CSS 464.0</li><li>■ CSS 465.0</li><li>■ CSS 466.0</li><li>■ CSS 467.0</li><li>■ CSS 468.0</li><li>■ CSS 469.0</li><li>■ CSS 470.0</li><li>■ CSS 471.0</li><li>■ CSS 472.0</li><li>■ CSS 473.0</li><li>■ CSS 474.0</li><li>■ CSS 475.0</li><li>■ CSS 476.0</li><li>■ CSS 477.0</li><li>■ CSS 478.0</li><li>■ CSS 479.0</li><li>■ CSS 480.0</li><li>■ CSS 481.0</li><li>■ CSS 482.0</li><li>■ CSS 483.0</li><li>■ CSS 484.0</li><li>■ CSS 485.0</li><li>■ CSS 486.0</li><li>■ CSS 487.0</li><li>■ CSS 488.0</li><li>■ CSS 489.0</li><li>■ CSS 490.0</li><li>■ CSS 491.0</li><li>■ CSS 492.0</li><li>■ CSS 493.0</li><li>■ CSS 494.0</li><li>■ CSS 495.0</li><li>■ CSS 496.0</li><li>■ CSS 497.0</li><li>■ CSS 498.0</li><li>■ CSS 499.0</li><li>■ CSS 500.0</li><li>■ CSS 501.0</li><li>■ CSS 502.0</li><li>■ CSS 503.0</li><li>■ CSS 504.0</li><li>■ CSS 505.0</li><li>■ CSS 506.0</li><li>■ CSS 507.0</li><li>■ CSS 508.0</li><li>■ CSS 509.0</li><li>■ CSS 510.0</li><li>■ CSS 511.0</li><li>■ CSS 512.0</li><li>■ CSS 513.0</li><li>■ CSS 514.0</li><li>■ CSS 515.0</li><li>■ CSS 516.0</li><li>■ CSS 517.0</li><li>■ CSS 518.0</li><li>■ CSS 519.0</li><li>■ CSS 520.0</li><li>■ CSS 521.0</li><li>■ CSS 522.0</li><li>■ CSS 523.0</li><li>■ CSS 524.0</li><li>■ CSS 525.0</li><li>■ CSS 526.0</li><li>■ CSS 527.0</li><li>■ CSS 528.0</li><li>■ CSS 529.0</li><li>■ CSS 530.0</li><li>■ CSS 531.0</li><li>■ CSS 532.0</li><li>■ CSS 533.0</li><li>■ CSS 534.0</li><li>■ CSS 535.0</li><li>■ CSS 536.0</li><li>■ CSS 537.0</li><li>■ CSS 538.0</li><li>■ CSS 539.0</li><li>■ CSS 540.0</li><li>■ CSS 541.0</li><li>■ CSS 542.0</li><li>■ CSS 543.0</li><li>■ CSS 544.0</li><li>■ CSS 545.0</li><li>■ CSS 546.0</li><li>■ CSS 547.0</li><li>■ CSS 548.0</li><li>■ CSS 549.0</li><li>■ CSS 550.0</li><li>■ CSS 551.0</li><li>■ CSS 552.0</li><li>■ CSS 553.0</li><li>■ CSS 554.0</li><li>■ CSS 555.0</li><li>■ CSS 556.0</li><li>■ CSS 557.0</li><li>■ CSS 558.0</li><li>■ CSS 559.0</li><li>■ CSS 560.0</li><li>■ CSS 561.0</li><li>■ CSS 562.0</li><li>■ CSS 563.0</li><li>■ CSS 564.0</li><li>■ CSS 565.0</li><li>■ CSS 566.0</li><li>■ CSS 567.0</li><li>■ CSS 568.0</li><li>■ CSS 569.0</li><li>■ CSS 570.0</li><li>■ CSS 571.0</li><li>■ CSS 572.0</li><li>■ CSS 573.0</li><li>■ CSS 574.0</li><li>■ CSS 575.0</li><li>■ CSS 576.0</li><li>■ CSS 577.0</li><li>■ CSS 578.0</li><li>■ CSS 579.0</li><li>■ CSS 580.0</li><li>■ CSS 581.0</li><li>■ CSS 582.0</li><li>■ CSS 583.0</li><li>■ CSS 584.0</li><li>■ CSS 585.0</li><li>■ CSS 586.0</li><li>■ CSS 587.0</li><li>■ CSS 588.0</li><li>■ CSS 589.0</li><li>■ CSS 590.0</li><li>■ CSS 591.0</li><li>■ CSS 592.0</li><li>■ CSS 593.0</li><li>■ CSS 594.0</li><li>■ CSS 595.0</li><li>■ CSS 596.0</li><li>■ CSS 597.0</li><li>■ CSS 598.0</li><li>■ CSS 599.0</li><li>■ CSS 600.0</li><li>■ CSS 601.0</li><li>■ CSS 602.0</li><li>■ CSS 603.0</li><li>■ CSS 604.0</li><li>■ CSS 605.0</li><li>■ CSS 606.0</li><li>■ CSS 607.0</li><li>■ CSS 608.0</li><li>■ CSS 609.0</li><li>■ CSS 610.0</li><li>■ CSS 611.0</li><li>■ CSS 612.0</li><li>■ CSS 613.0</li><li>■ CSS 614.0</li><li>■ CSS 615.0</li><li>■ CSS 616.0</li><li>■ CSS 617.0</li><li>■ CSS 618.0</li><li>■ CSS 619.0</li><li>■ CSS 620.0</li><li>■ CSS 621.0</li><li>■ CSS 622.0</li><li>■ CSS 623.0</li><li>■ CSS 624.0</li><li>■ CSS 625.0</li><li>■ CSS 626.0</li><li>■ CSS 627.0</li><li>■ CSS 628.0</li><li>■ CSS 629.0</li><li>■ CSS 630.0</li><li>■ CSS 631.0</li><li>■ CSS 632.0</li><li>■ CSS 633.0</li><li>■ CSS 634.0</li><li>■ CSS 635.0</li><li>■ CSS 636.0</li><li>■ CSS 637.0</li><li>■ CSS 638.0</li><li>■ CSS 639.0</li><li>■ CSS 640.0</li><li>■ CSS 641.0</li><li>■ CSS 642.0</li><li>■ CSS 643.0</li><li>■ CSS 644.0</li><li>■ CSS 645.0</li><li>■ CSS 646.0</li><li>■ CSS 647.0</li><li>■ CSS 648.0</li><li>■ CSS 649.0</li><li>■ CSS 650.0</li><li>■ CSS 651.0</li><li>■ CSS 652.0</li><li>■ CSS 653.0</li><li>■ CSS 654.0</li><li>■ CSS 655.0</li><li>■ CSS 656.0</li><li>■ CSS 657.0</li><li>■ CSS 658.0</li><li>■ CSS 659.0</li><li>■ CSS 660.0</li><li>■ CSS 661.0</li><li>■ CSS 662.0</li><li>■ CSS 663.0</li><li>■ CSS 664.0</li><li>■ CSS 665.0</li><li>■ CSS 666.0</li><li>■ CSS 667.0</li><li>■ CSS 668.0</li><li>■ CSS 669.0</li><li>■ CSS 670.0</li><li>■ CSS 671.0</li><li>■ CSS 672.0</li><li>■ CSS 673.0</li><li>■ CSS 674.0</li><li>■ CSS 675.0</li><li>■ CSS 676.0</li><li>■ CSS 677.0</li><li>■ CSS 678.0</li><li>■ CSS 679.0</li><li>■ CSS 680.0</li><li>■ CSS 681.0</li><li>■ CSS 682.0</li><li>■ CSS 683.0</li><li>■ CSS 684.0</li><li>■ CSS 685.0</li><li>■ CSS 686.0</li><li>■ CSS 687.0</li><li>■ CSS 688.0</li><li>■ CSS 689.0</li><li>■ CSS 690.0</li><li>■ CSS 691.0</li><li>■ CSS 692.0</li><li>■ CSS 693.0</li><li>■ CSS 694.0</li><li>■ CSS 695.0</li><li>■ CSS 696.0</li><li>■ CSS 697.0</li><li>■ CSS 698.0</li><li>■ CSS 699.0</li><li>■ CSS 700.0</li><li>■ CSS 701.0</li><li>■ CSS 702.0</li><li>■ CSS 703.0</li><li>■ CSS 704.0</li><li>■ CSS 705.0</li><li>■ CSS 706.0</li><li>■ CSS 707.0</li><li>■ CSS 708.0</li><li>■ CSS 709.0</li><li>■ CSS 710.0</li><li>■ CSS 711.0</li><li>■ CSS 712.0</li><li>■ CSS 713.0</li><li>■ CSS 714.0</li><li>■ CSS 715.0</li><li>■ CSS 716.0</li><li>■ CSS 717.0</li><li>■ CSS 718.0</li><li>■ CSS 719.0</li><li>■ CSS 720.0</li><li>■ CSS 721.0</li><li>■ CSS 722.0</li><li>■ CSS 723.0</li><li>■ CSS 724.0</li><li>■ CSS 725.0</li><li>■ CSS 726.0</li><li>■ CSS 727.0</li><li>■ CSS 728.0</li><li>■ CSS 729.0</li><li>■ CSS 730.0</li><li>■ CSS 731.0</li><li>■ CSS 732.0</li><li>■ CSS 733.0</li><li>■ CSS 734.0</li><li>■ CSS 735.0</li><li>■ CSS 736.0</li><li>■ CSS 737.0</li><li>■ CSS 738.0</li><li>■ CSS 739.0</li><li>■ CSS 740.0</li><li>■ CSS 741.0</li><li>■ CSS 742.0</li><li>■ CSS 743.0</li><li>■ CSS 744.0</li><li>■ CSS 745.0</li><li>■ CSS 746.0</li><li>■ CSS 747.0</li><li>■ CSS 748.0</li><li>■ CSS 749.0</li><li>■ CSS 750.0</li><li>■ CSS 751.0</li><li>■ CSS 752.0</li><li>■ CSS 753.0</li><li>■ CSS 754.0</li><li>■ CSS 755.0</li><li>■ CSS 756.0</li><li>■ CSS 757.0</li><li>■ CSS 758.0</li><li>■ CSS 759.0</li><li>■ CSS 760.0</li><li>■ CSS 761.0</li><li>■ CSS 762.0</li><li>■ CSS 763.0</li><li>■ CSS 764.0</li><li>■ CSS 765.0</li><li>■ CSS 766.0</li><li>■ CSS 767.0</li><li>■ CSS 768.0</li><li>■ CSS 769.0</li><li>■ CSS 770.0</li><li>■ CSS 771.0</li><li>■ CSS 772.0</li><li>■ CSS 773.0</li><li>■ CSS 774.0</li><li>■ CSS 775.0</li><li>■ CSS 776.0</li><li>■ CSS 777.0</li><li>■ CSS 778.0</li><li>■ CSS 779.0</li><li>■ CSS 780.0</li><li>■ CSS 781.0</li><li>■ CSS 782.0</li><li>■ CSS 783.0</li><li>■ CSS 784.0</li><li>■ CSS 785.0</li><li>■ CSS 786.0</li><li>■ CSS 787.0</li><li>■ CSS 788.0</li><li>■ CSS 789.0</li><li>■ CSS 790.0</li><li>■ CSS 791.0</li><li>■ CSS 792.0</li><li>■ CSS 793.0</li><li>■ CSS 794.0</li><li>■ CSS 795.0</li><li>■ CSS 796.0</li><li>■ CSS 797.0</li><li>■ CSS 798.0</li><li>■ CSS 799.0</li><li>■ CSS 800.0</li><li>■ CSS 801.0</li><li>■ CSS 802.0</li><li>■ CSS 803.0</li><li>■ CSS 804.0</li><li>■ CSS 805.0</li><li>■ CSS 806.0</li><li>■ CSS 807.0</li><li>■ CSS 808.0</li><li>■ CSS 809.0</li><li>■ CSS 810.0</li><li>■ CSS 811.0</li><li>■ CSS 812.0</li><li>■ CSS 813.0</li><li>■ CSS 814.0</li><li>■ CSS 815.0</li><li>■ CSS 816.0</li><li>■ CSS 817.0</li><li>■ CSS 818.0</li><li>■ CSS 819.0</li><li>■ CSS 820.0</li><li>■ CSS 821.0</li><li>■ CSS 822.0</li><li>■ CSS 823.0</li><li>■ CSS 824.0</li><li>■ CSS 825.0</li><li>■ CSS 826.0</li><li>■ CSS 827.0</li><li>■ CSS 828.0</li><li>■ CSS 829.0</li><li>■ CSS 830.0</li><li>■ CSS 831.0</li><li>■ CSS 832.0</li><li>■ CSS 833.0</li><li>■ CSS 834.0</li><li>■ CSS 835.0</li><li>■ CSS 836.0</li><li>■ CSS 837.0</li><li>■ CSS 838.0</li><li>■ CSS </li></ul>		

## Speaker notes

Die Web-Plattform umfasst eine ausserordentlich grosse Zahl von Technologien, APIs, Frameworks und Bibliotheken. Dazu kommt, dass sich das ganze Gebiet schnell entwickelt.

Hier einen vernünftigen Auschnitt zu definieren und in einem Semester zu vermitteln, ist eine Herausforderung. Wir versuchen unser Bestes ... :)

- <https://html-now.github.io>
- <https://caniuse.com/#index>

# WEB OMNIPRÄSENT

- Web-Seiten oder Web-Apps im Browser
- Auf Desktop, Notebook, oder Smartphone
- Native Mobilapps mit Web-Kern (PhoneGap, Cordova)
- Native Desktop-Apps auf Web-Basis (Electron)
- Apps, welche REST-basierte Webservices nutzen

## Speaker notes

Webtechnologien findet man heute überall. Klar, zunächst einmal Webangeboten, die man im Browser auf dem Desktop-PC oder dem Notebook, ebenso wie auf dem Smartphone, nutzt. Aber auch in nativen Applikationen steckt häufig Web-Technologie. Sei es, dass die ganze App als Web-Applikation entwickelt wurde und dann in eine native Hülle gesteckt wurde (PhoneGap, Cordova), sei es, dass nur ein Teil einer App eine WebView verwendet, etwa für das Hilfesystem. Applikationen wie VSCode verwenden mit Electron ein Plattform basierend auf Chromium und Node.js.

- <https://phonegap.com>
- <https://cordova.apache.org>
- <https://www.electronjs.org>

# ZIELE WBE (WH)

- JavaScript gut verstehen und einsetzen können
- Nebenbei: HTML und CSS verstehen (Selbststudium)
- Auswahl von Web-APIs kennen und einsetzen können
- Eigenes kleines Web-Framework erstellen können

# ABGRENZUNG

- Schwerpunkt Front-end, wenig Back-end
- Beschränkung auf ausgewählte Standards und Web-APIs
- Nur ansatzweise: mobiles Web und zugehörige APIs
- Kein Thema: Gestaltung, Usability und Accessibility

## Speaker notes

Einige Themen müssen explizit aussen vor bleiben, obwohl sie wichtige Themen für Web-Entwickler sind.

*Back-end:* Bis auf wenige Ausnahmen werden wir uns vor allem mit dem Front-end beschäftigen. Die Backend-Entwicklung ist selbst ein riesiges Gebiet und würde Stoff für mindestens ein weiteres Semester umfassen.

*Mobiles Web:* Ein grosser Teil der Web-Nutzung entfällt heute auf Smartphones. Wie Webseiten und Anwendungen so umgesetzt werden, dass sie auf verschiedenen Gerätetypen genutzt werden können, werden wir ansehen. Eine vertiefte Behandlung mobil-spezifischer Themen ist im Rahmen von WBE aber nicht möglich, etwa die Entwicklung hybrider Apps (PhoneGap/Cordova) oder von Progressive Web Apps (PWAs). Hier müssen wir auf das Wahlfach "Mobile Applications" verweisen, in dem neben dem mobilen Web auch native Apps für Android und iOS entwickelt werden.

*Web-APIs:* Nur eine kleine Zahl der WEB-APIs können wir im Rahmen von WBE ansehen, viele interessante APIs bleiben leider unberücksichtigt.

*Gestaltung, Usability und Accessibility:* Das sind ebenfalls sehr wichtige Themen, speziell für Front-end-Entwickler. Leider reicht auch für diese Themen die Zeit in WBE nicht.

Auch wenn die oben genannten Themen in WBE nicht vertieft behandelt werden können, wird es doch für Interessierte zum einen oder anderen Thema Tipps, Quellen und Material für die Einarbeitung geben.

# FRONT-END DEVELOPMENT

- A practical guide to learning front end development for beginners  
<https://www.freecodecamp.org/news/a-practical-guide-to-learning-front-end-development-for-beginners-da6516505e41/>
- Developer Roadmap  
<https://github.com/kamranahmedse/developer-roadmap>
- Learn to become a modern Frontend Developer in 2020  
<https://medium.com/@kamranahmedse/modern-frontend-developer-in-2018-4c2072fa2b9c>
- MDN: Learn Web Development  
<https://developer.mozilla.org/en-US/docs/Learn>
- Frontend Masters Books  
<https://frontendmasters.com/guides/>



