



openHPI: **Web-Technologien** Kursresumé

Prof. Dr. Christoph Meinel

Hasso-Plattner-Institut

Universität Potsdam

World Wide Web ist **verteiltes Hypermedia-System**, untereinander verlinkte Web-Dokumente liegen weltweit auf verschiedenen Servern

- Zum Abruf der Dokumente wird eindeutiges Adressierungsschema benötigt
 - Jedes Dokument braucht (mindestens) einen **Uniform Resource Identifier – URI**
 - URI meist als **Uniform Resource Locator – URL** – realisiert, Identifikation von Ressourcen erfolgt über deren „Adresse“ (Domain Name des Server, Pfad auf dem Server, Dateiname)
- Anforderung und Übertragung der Dokumente erfolgt mit **Hypertext Transfer Protocol – HTTP**
 - HTTP bietet einfaches Frage-Antwort-Schema (Request / Response), mit dem Clients (Browser) bei Servern Dokumente anfordern können

World Wide Web ist **verteiltes Hypermedia-System**, untereinander verlinkte Web-Dokumente liegen weltweit auf verschiedenen Servern

- Erweiterungen für HTTP verbessern Performance:
 - **Persistente Verbindungen** und Pipelining
 - **Kompression**
 - **Caching** (in Clients, Servern und Zwischensystemen – Proxies)
- Überwindung der Zustandslosigkeit von HTTP mittels **Sessions und Cookies** für Web-Anwendungen möglich
- Browser und Server können Auslieferung von Content-Varianten aushandeln – **Content Negotiation**
- Nach über 25 Jahren wird mit **HTTP/2** derzeit Nachfolger für das zentrale Web-Protokoll eingeführt

Dokumente im WWW werden mit HTML und CSS beschrieben

- Trennung von Struktur und Gestaltung
 - **Hypertext Markup Language – HTML** – für strukturierte Aufbereitung des Dokumenteninhalts
 - **Cascading Stylesheets – CSS** – für Gestaltung der Darstellung von Dokumenten auf verschiedenen Ausgabegeräten
- HTML führt **Hyperlink**-Konzept ein
- HTML-Dokumente bestehen aus
 - Head: Informationen *über* den Inhalt (Metainformationen)
 - Body: eigentlicher Inhalte
- HTML ist **Markup-Sprache**: Strukturgebende Markups („Tags“) sind Teil des Dokumenteninhalts und werden durch Markup-Trenner (<...>) gekennzeichnet

Dokumente im WWW werden mit HTML und CSS beschrieben

- Aktuelle Version **HTML 5** führt „sprechende“ Strukturelemente, Multimedia-Tags, neue Formular-Funktionen und Zeichenfläche (Canvas) ein
- CSS-Regeln legen fest, welche HTML-Elemente wie gestaltet werden sollen
 - **Selektoren** wählen zu gestaltendes Element aus
 - **Deklarationen** weisen bestimmten Gestaltungsmerkmalen einen Wert zu
- **Responsives Webdesign** sorgt dafür, dass Webseiten automatisch an spezifische Anforderungen der verschiedenen Endgeräte (Desktop, Smartphone, Tablet, ...) angepasst werden
- **Extensible Markup Language – XML** ist (wie SGML) Meta-Markupsprache, also Sprache zur Beschreibung von Sprachen

Komplexere Nutzerinteraktion im Web und Web-Anwendungen werden mittels Web-Programmierung realisiert

- Wir unterscheiden **clientseitige** und **serverseitige** Web-Programmierung
- Bei clientseitiger Web-Programmierung wird Programmcode mit Webseite ausgeliefert und dann **im Browser ausgeführt**
- **Document Object Model – DOM** – bietet Schnittstelle für die Manipulation von HTML-Elementen
- Bei serverseitiger Web-Programmierung wird **Programm auf dem Server ausgeführt** und Ausgabe (z.B. HTML-Dokument) an Browser zur Anzeige ausgeliefert
- Für **Kodierung von Informationen** (Text, Grafik, Multimedia) zur Übermittlung im Web gibt es spezifische Anforderungen und Formate

JavaScript ist das wichtigste Werkzeug für clientseitige Web-Programmierung

- JavaScript (standardisiert als ECMAScript) ist Skriptsprache zur Manipulation von HTML-Elementen
- Für viele gängige Anwendungsfälle gibt es umfangreiche **Bibliotheken** und **Frameworks**, z.B. jQuery oder D3
- Auch für CSS gibt es Reihe an Frameworks, die wiederkehrende Aufgaben vereinfachen
 - **CSS-Präprozessoren** erweitern CSS-Syntax, z.B. um Variablen und Funktionen
- **AJAX – Asynchronous JavaScript and XML** – erlaubt Kommunikation zwischen clientseitiger Web-Anwendung und Servern, z.B. zur hintergründigen Anfrage zusätzlicher Ressourcen

Viele Aufgaben der Web-Programmierung – insbesondere solche, bei denen große Datenmengen zu verarbeiten sind – werden auf dem Server gelöst

- Webserver ruft **serverseitiges Programm** auf (z.B. über **CGI-Schnittstelle**) und übergibt Anfrageparameter aus dem HTTP-Request
- Programm auf dem Server wird ausgeführt, erzeugt **parameterabhängige Ausgabe**, z.B. als **HTML** oder **JSON**, und gibt diese an Webserver zurück (der diese als HTTP-Response an Client ausliefert)
- Serverseitige Web-Programmierung basiert heute fast immer auf mächtigen **Web-Frameworks**, z.B. Ruby on Rails oder Django

Viele Aufgaben der Web-Programmierung – insbesondere solche, bei denen große Datenmengen verarbeitet werden – werden auf dem Server gelöst

- Daten auf dem Server werden **in Datenbanken persistiert** und über Web-Framework mit **Object Relational Mapper – ORM** – abgefragt
 - **Relationale Datenbanken:** Fixe Datenstruktur (Tabellen), hohe Datenkonsistenz
 - **NoSQL-Datenbanken:** Flexible Datenstruktur, leichter zu skalieren, Konsistenz schwieriger zu gewährleisten
- **Web Services** ermöglichen **Kommunikation zwischen** unterschiedlichen **Web-Anwendungen** und erlauben so Realisierung verteilter Applikationen

Soziale Netzwerke, das Semantic Web und Anwendungen für Web Services (z.B. im IoT Bereich) sind wichtige Themen für das Web von heute und morgen

- **Soziale Netzwerke** und **Social Media** sind aktuelle Killer-Applikationen des Webs
- Kritisch zu betrachten sind trotz oder gerade wegen der weitgehenden Verbreitung
 - Manipulationsmöglichkeiten bei der Meinungsbildung
 - Implikationen für Datenschutz und Sicherheit
- **Semantisches Web** verknüpft nicht länger nur Dokumente, sondern auch **Dinge** (Menschen, Orten, Ereignissen, ...) und setzt sie in **Beziehung**
 - **Bedeutung** einer Information wird auch erkennbar für Maschine

Soziale Netzwerke, das Semantic Web und Anwendungen für Web Services (z.B. im IoT-Bereich) sind wichtige Themen für das Web von heute und morgen

- **Internet of Things:** „Dinge“ (physikalische Geräte, Sensoren, ...) erhalten „digitale Hülle“ – können über Entfernung angesprochen werden oder auch miteinander kommunizieren
- **Smart Home, Smart Factory, Smart Traffic, ...** zeigen Visionen auf für mögliche Veränderungen dank digitaler Technologien
- **Sicherheit** und **Datenschutz** sind zentrale Herausforderungen für tiefergehende Verankerung der Smart-Technologien in der Gesellschaft

Wir verabschieden uns und ...

- ... hoffen, Sie hatten Freude im Kurs und konnten etwas dazulernen
- Alle Kursinhalte stehen auch nach Ende des Kurses weiter zur Verfügung



Prof. Dr.
Christoph Meinel



Christian
Willems



Matthias
Bauer



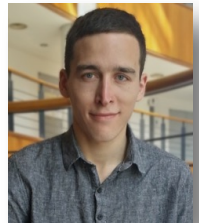
Martin
Malchow



Thomas
Staubitz



Tobias
Rohloff



Franz
Liedke

Wir wünschen Ihnen viel Erfolg bei der Abschlussprüfung!