

Hochschule Darmstadt

Fachbereich Informatik

Entwicklung webbasierter Anwendungen



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Hochschule Darmstadt

Fachbereich Informatik

Wiederholung



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

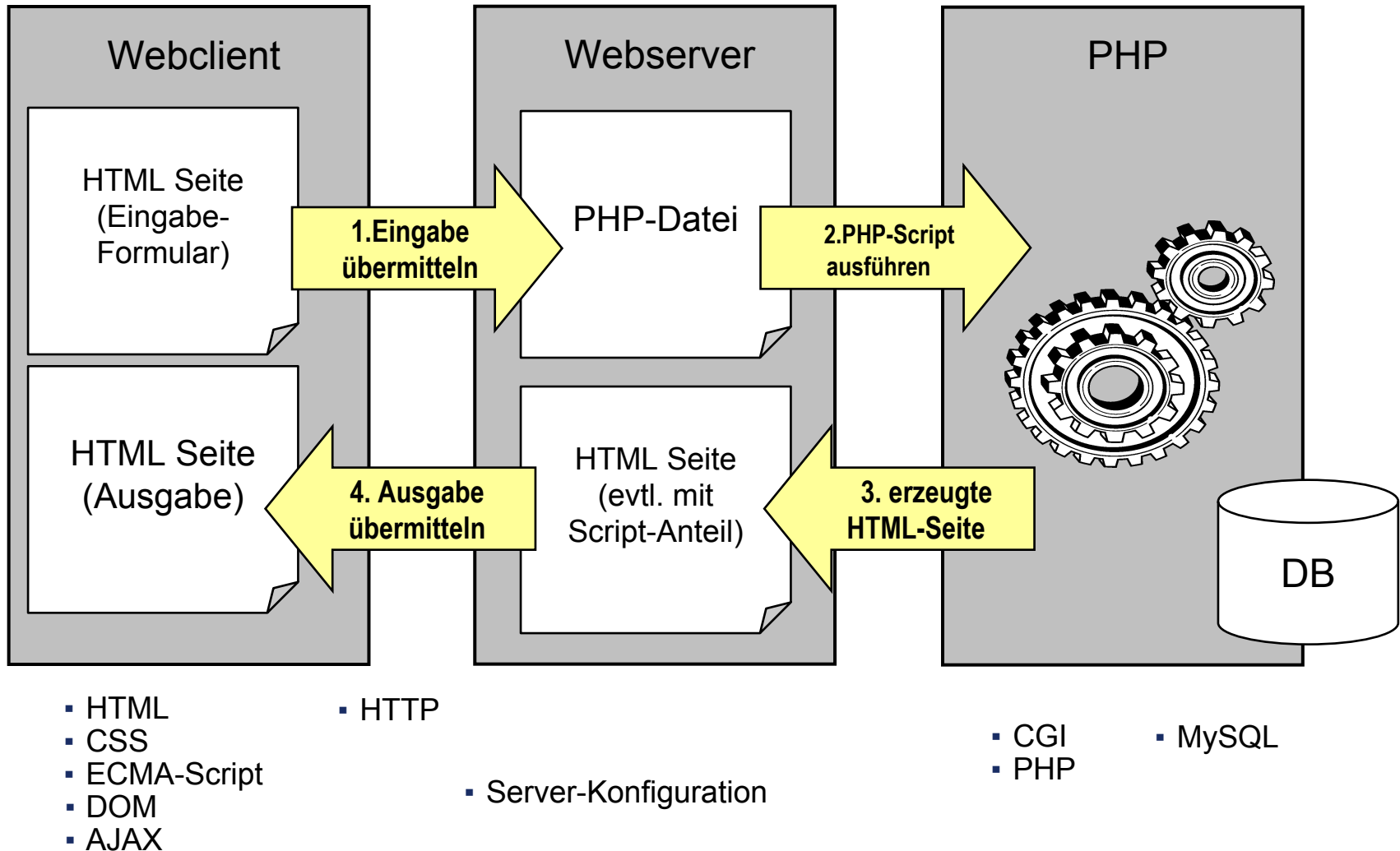
fbi

FACHBEREICH INFORMATIK

1. Einleitung

Einsatz der Technologien im Zusammenhang

Es ist enorm wichtig zu verstehen, welche Inhalte in welchem Kontext erzeugt werden

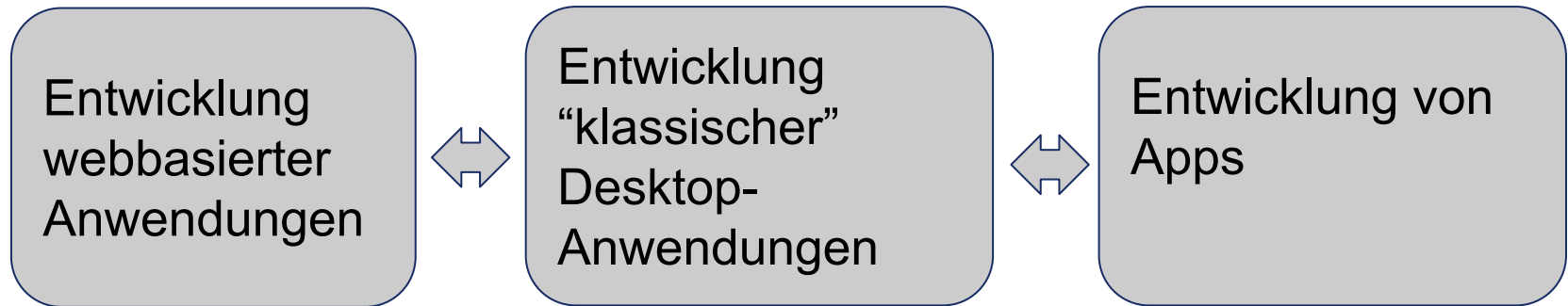


Anforderungsanalyse: nicht-funktionale Anforderungen

- Es gibt im Internet ein riesiges Angebot
 - meine Anwendung ist nur eine unter vielen
 - Benutzer wechseln häufig die Anwendungen / Sites
- Benutzer scannen statt zu lesen
 - 79% überfliegen die Seiten nur
 - Schulung darf absolut nicht erforderlich sein;
 - Hilfesystem muss überflüssig sein
- Viele unerfahrene Benutzer
 - Kinder, Senioren
- Unterschiedliche Systeme der Benutzer
 - Browser, Plugins, CPU, Bildschirme, Datenverbindung
- Performanz
 - Support für verschiedene Browser, Ausgabegeräte, Transferraten,
 - Anzahl der Benutzer, Häufigkeit des Datenaustauschs,...
- Darstellung
 - Stil, Corporate Identity, Farbschema

mobile / responsive

Abgrenzung zu anderen Entwicklungs-Arten



- Professionelle Softwareentwicklung unterscheidet sich nicht

Agenda

- 1. Einleitung
 - 1.2 Ergonomie von webbasierten Anwendungen
- 2. Webclient
 - 2.1 HTML

Hochschule Darmstadt

Fachbereich Informatik

1.2 Ergonomie für webbasierte Anwendungen



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Mensch-Maschine-Schnittstelle

- Der Systemzustand muss auf einen Blick erkennbar sein (ohne ihn dabei zu verändern!)
 - ⇒ Grund-Forderung aus der Software-Ergonomie
- Wo bin ich ? Ort
 - ⇒ momentaner Aufenthaltsort im System
- Was kann ich hier tun ? Modus
 - ⇒ zur Verfügung stehende Operationen
- Wie kam ich hierher ? Weg
 - ⇒ Vorgeschichte, Kontext
- Wohin kann ich gehen ? Weg
 - ⇒ Ziel eines Verweises soll erkennbar sein



http://www.wpgs.de/images/stories/Lehrtexte_Abbildungen/Einfuehrung_Wirtschaftspsychologie/hmi.jpg

Es folgen Überlegungen, wie diese Fragen durch Layout- und Gestaltungselemente beantwortet werden können.



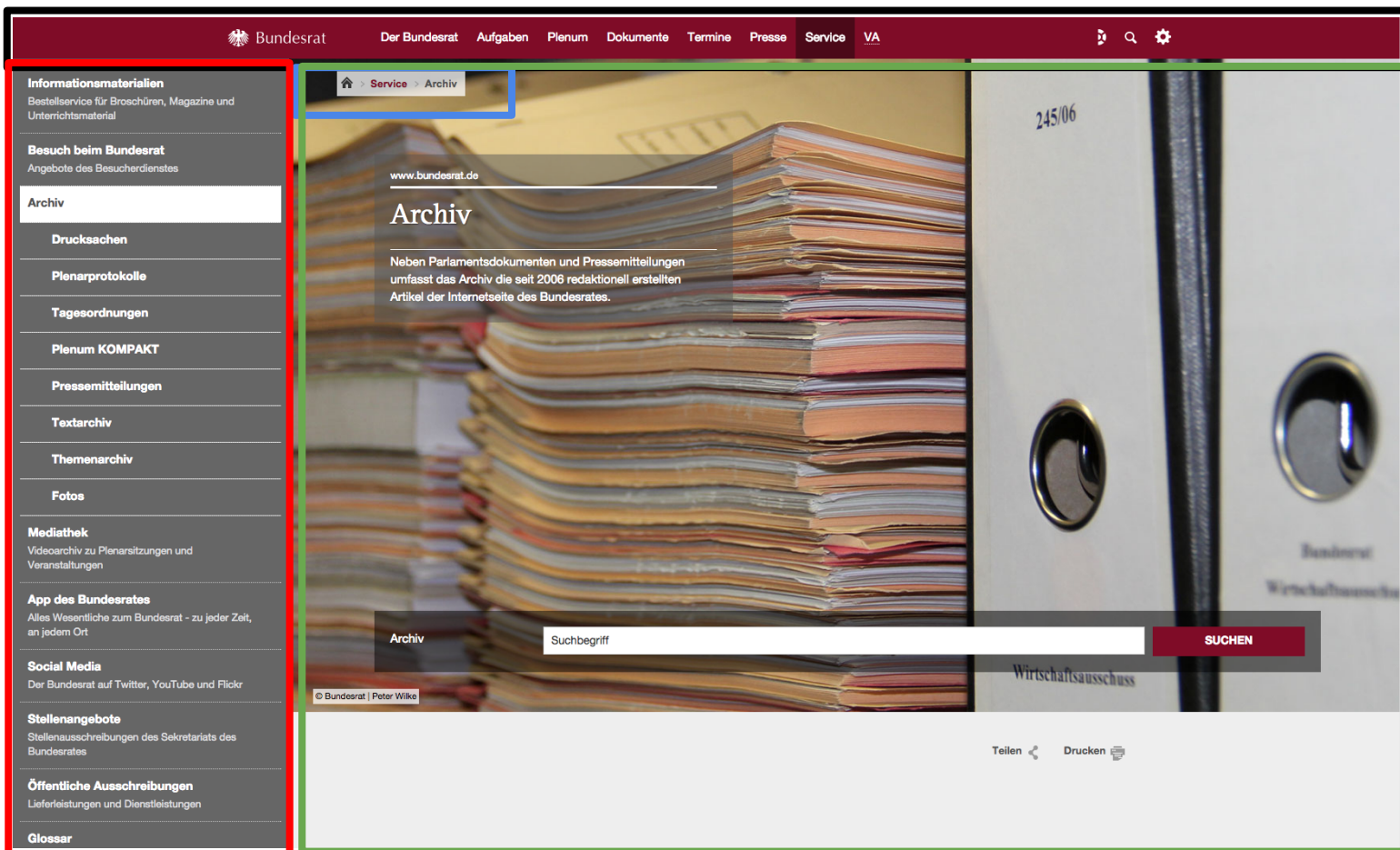
Layout: Komponenten einer Bildschirmseite

- **Orientierungskomponente** wo bin ich?
 - ⇒ dient der Orientierung des Benutzers
- **Präsentationskomponente** was kann ich hier tun?
 - ⇒ Darstellung der Informationsgehalts
 - ⇒ Bühne für Animationen, Simulationen, Videos
- **Navigationskomponente (Interaktions-)** wohin kann ich gehen?
 - ⇒ dient der Steuerung durch den Benutzer
- **Hintergrund**
 - ⇒ passives Designelement
 - ⇒ unverändert über mehrere Bildschirmseiten



1.2 Ergonomie für webbasierte Anwendungen

Layoutbeispiel: Bundesrat



Hintergrund

Orientierung

Präsentation

Navigation



- Webseiten sollen auch für Menschen mit einer Behinderung zugänglich sein
 - ⇒ Sehschwächen (z.B. Rot-Grün-Blindheit) aber auch Hör-, Lern-, Lese-, motorische Schwächen uvm.
 - ⇒ insbesondere soll eine Webseite vorgelesen werden können
 - Screenreader lesen die Seite mit Text2Speech vor
 - Braille-Zeilen geben Blindenschrift zeilenweise aus
 - Tabellen werden von links nach rechts und von oben nach unten gelesen
 - Bilder und Videos sind als solche nicht darstellbar
 - Der Inhalt muss logisch gruppiert und angeordnet sein – und nicht nach der Anordnung auf dem Bildschirm
 - ⇒ Die Bedienung muss mit vereinfachten Tastaturen möglich sein



Textausgabe mit Braille-Zeile
Bild: SBV



www.computer-fuer-behinderte.de

Hochschule Darmstadt

Fachbereich Informatik

2. Webclient



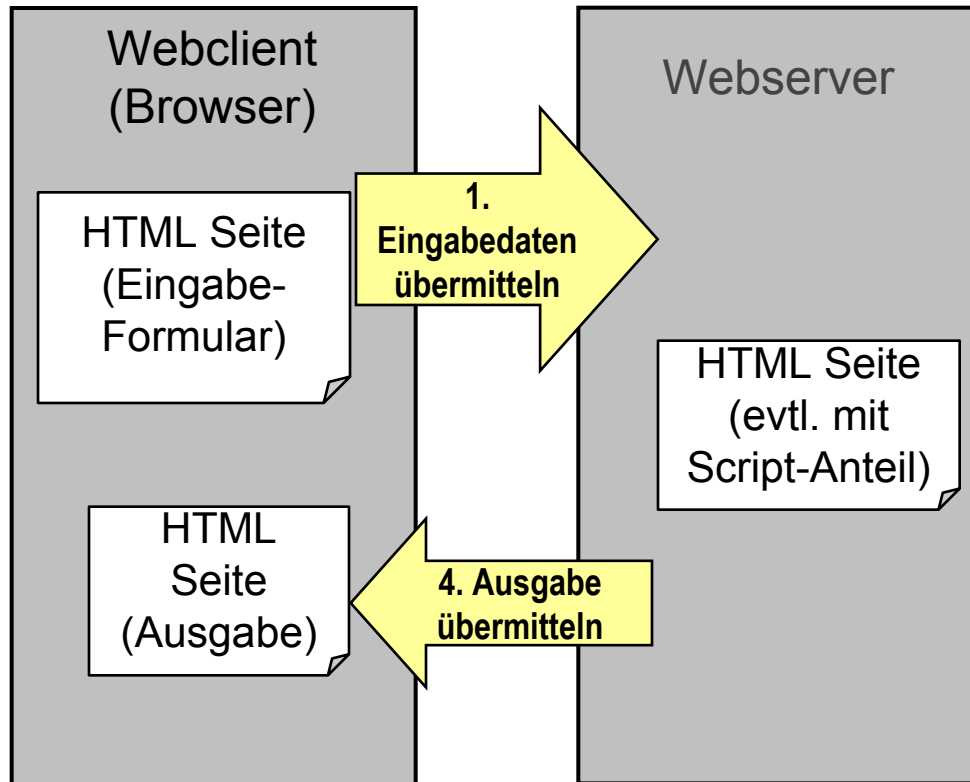
h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

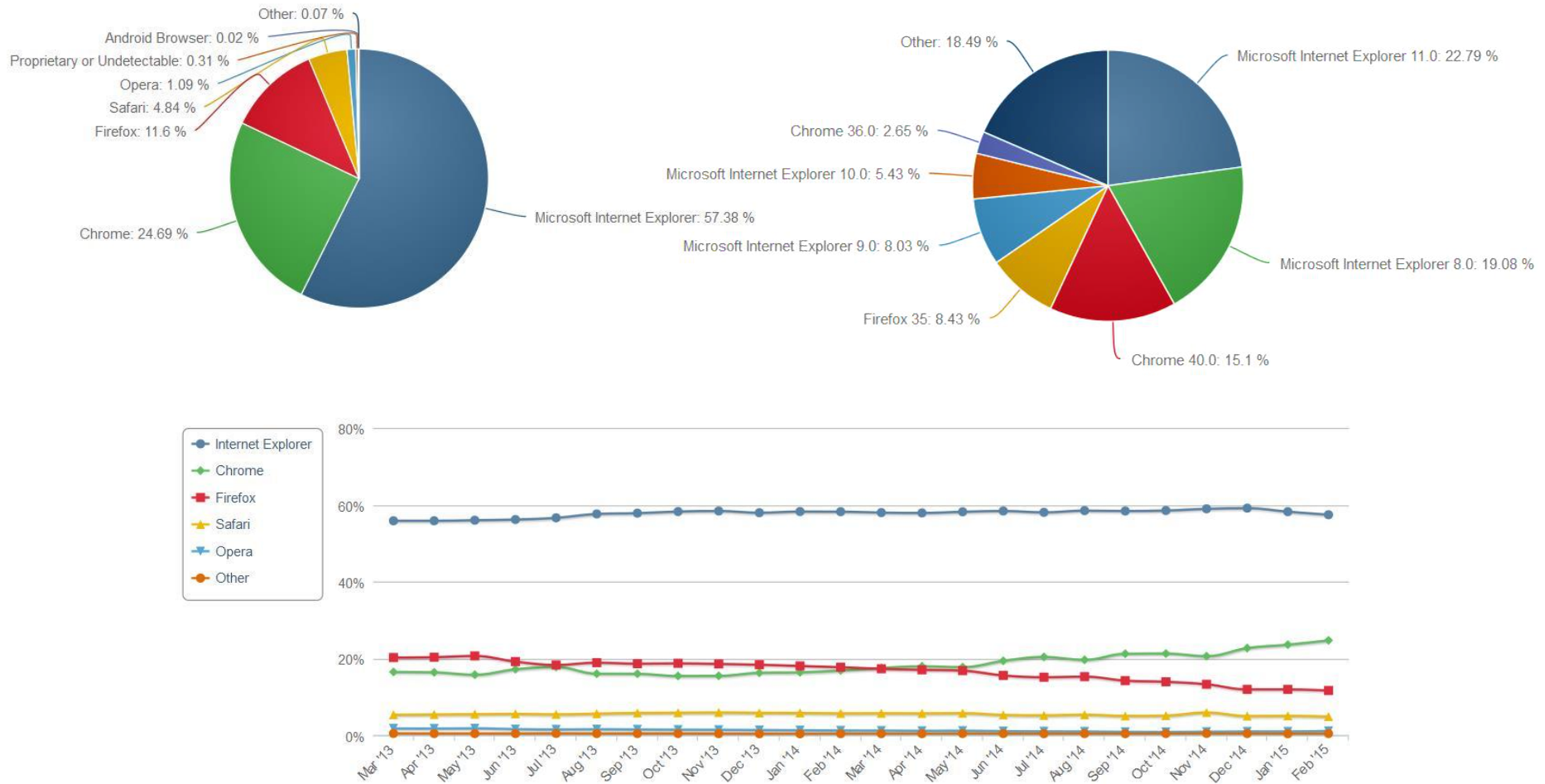
Der Webclient



- HTML
- CSS
- ECMA-Script
- DOM
- AJAX

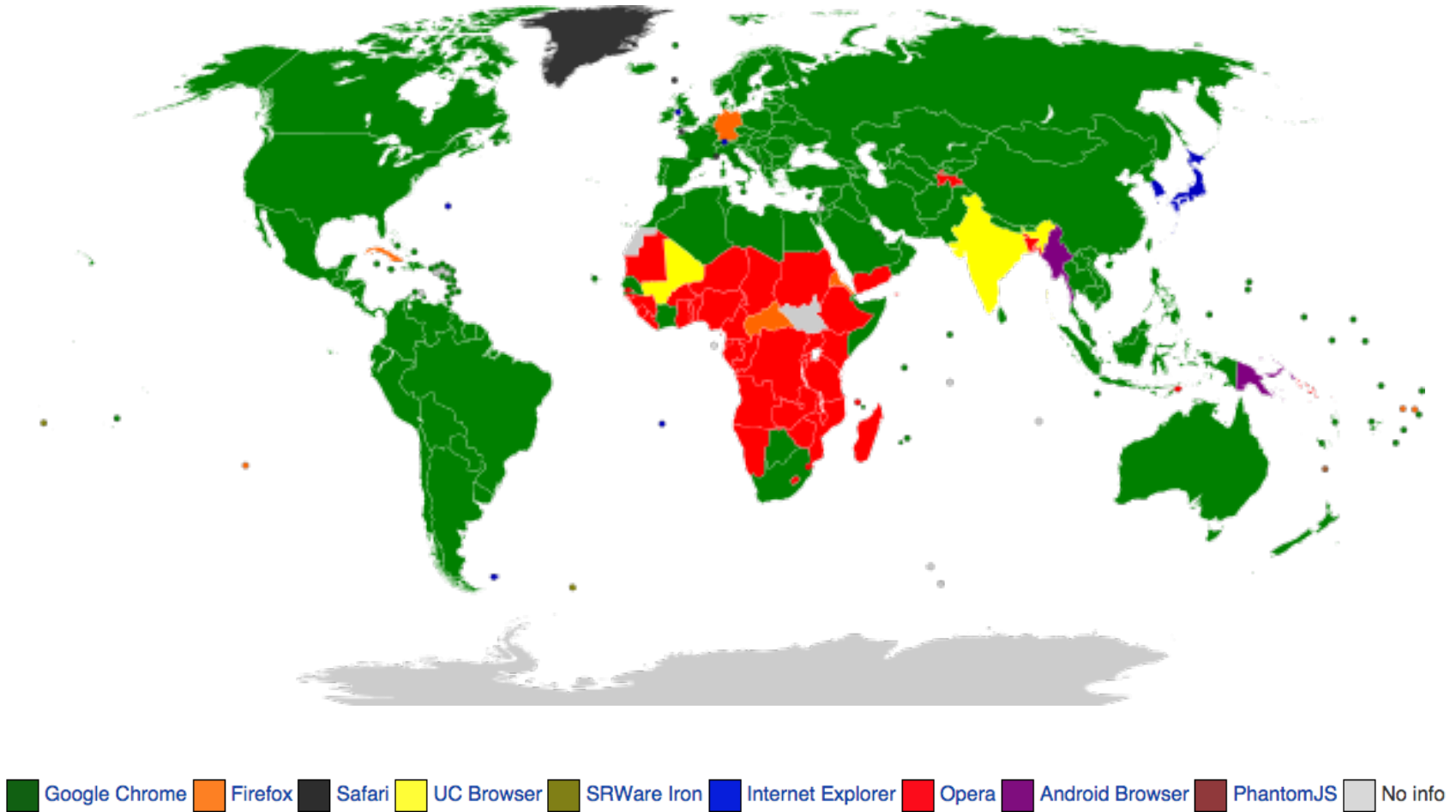
2. Webclient

Marktanteile und Trend der Browser für Desktops



Quelle: Net Applications, <http://marketshare.hitslink.com>, (Stand 03/2015)

Marktanteile und Trend der Browser für Desktops

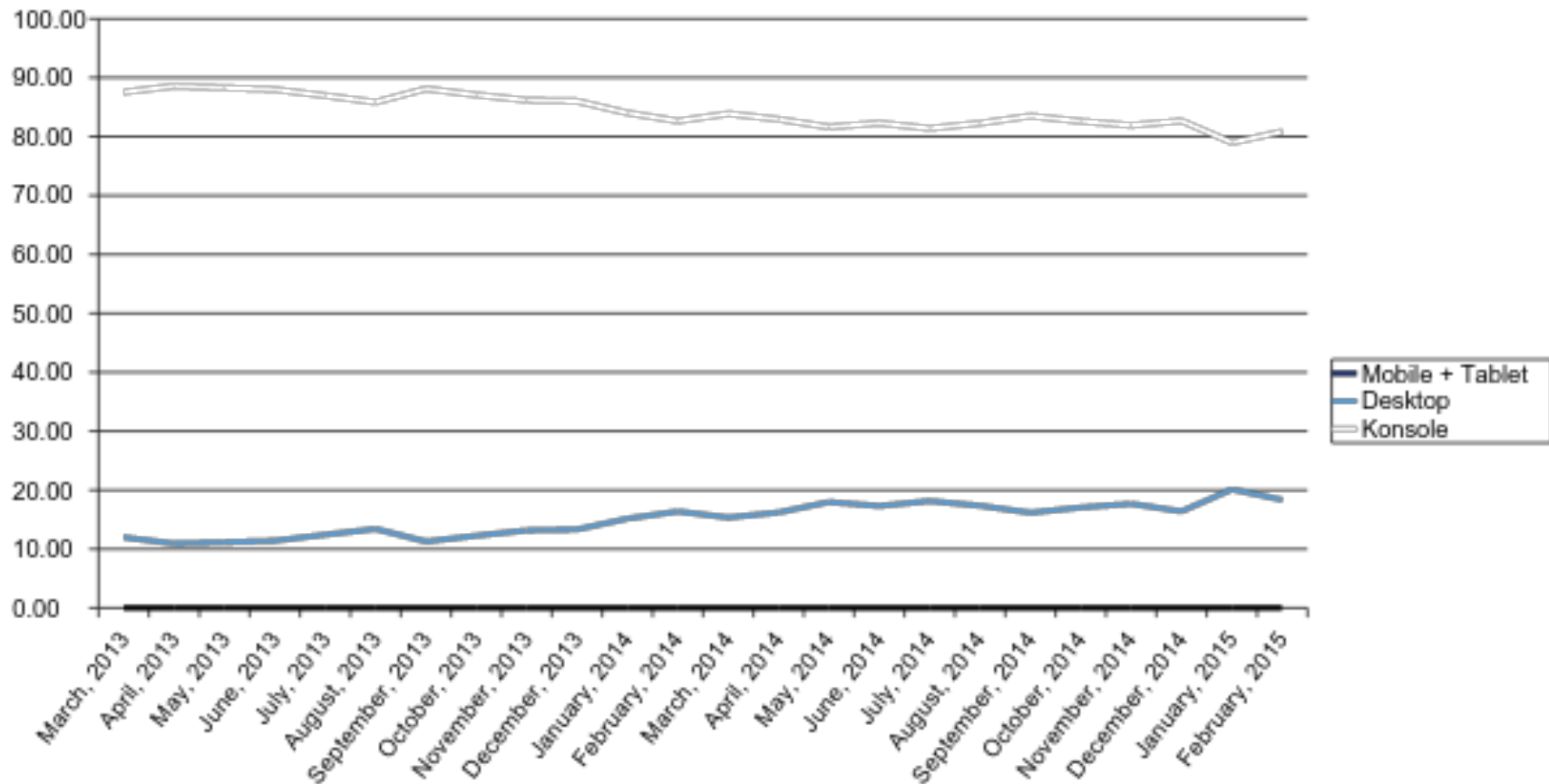


<http://gs.statcounter.com/#all-browser-ww-monthly-201506-201506-map> (Juni 2015)



2. Webclient

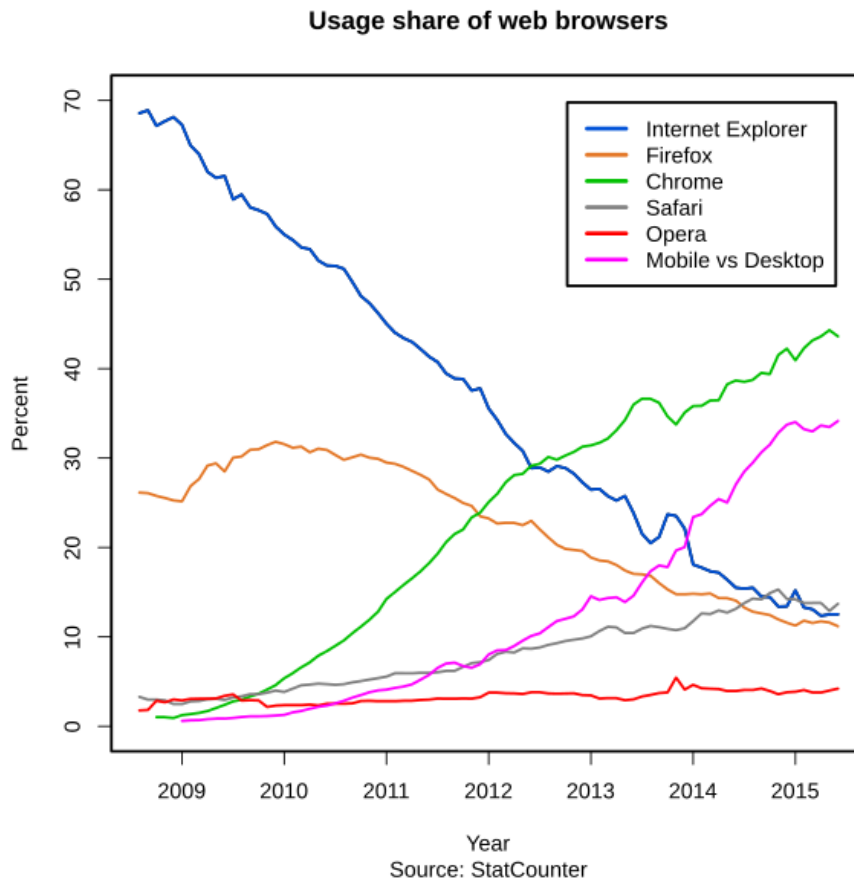
Browseranteil nach Endgerätetyp



Quelle: Net Applications, <http://marketshare.hitslink.com>, (Stand 03/2015)

Ca. 20% der Internetzugriffe kommt 2015 von mobilen Geräten.
Die Tendenz ist stark steigend. 2011 waren es nur 4%.

Marktanteile und Trend der Web-browser



mobile / responsive

Source: https://upload.wikimedia.org/wikipedia/commons/thumb/8/86/Usage_share_of_web_browsers_%28Source_StatCounter%29.svg/600px-Usage_share_of_web_browsers_%28Source_StatCounter%29.svg.png

Hochschule Darmstadt

Fachbereich Informatik

2.1 HTML



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Was ist HTML ?

```
<title> Text des Titels </title>
```

■ Markup Language

- ⇒ Markup Language: Auszeichnungssprache
- ⇒ markiert und attributiert Bestandteile eines Fließtexts
- ⇒ Layout bewusst nicht definiert
- ⇒ Browser setzen Auszeichnung in visuelle Darstellung um

■ HyperText: Verweise auf andere Dokumente

- ⇒ komfortable Querverweise zu anderen Stellen im eigenen Projekt oder zu beliebigen anderen Dokumenten im Web
- ⇒ URL (Uniform Resource Locator)

■ Text-Dateien (kein Binärformat)

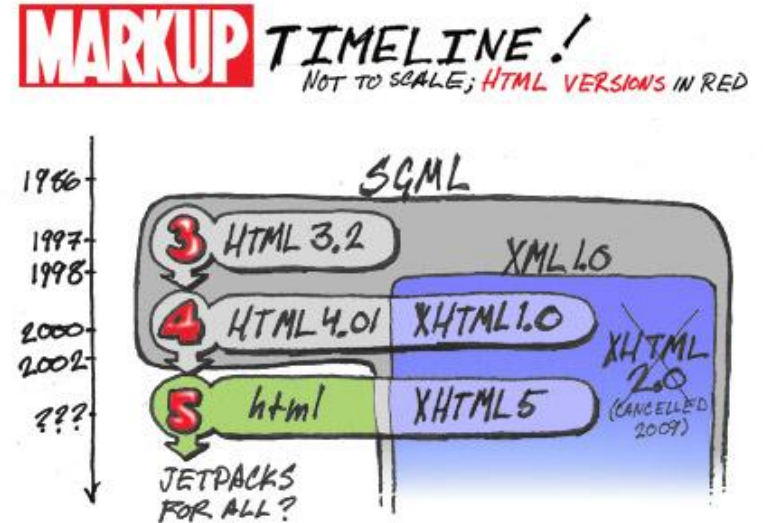
- ⇒ mit jedem Texteditor zu bearbeiten
- ⇒ leicht zu generieren und zu debuggen

HTML heißt HyperText Markup Language

2.1 HTML

SGML – XML – HTML - XHTML

- SGML (Standard Generalized Markup Language)
 - ⇒ definiert seit 1986 eine Sprachfamilie zur Auszeichnung von Text (ISO-Norm 8879)
 - ⇒ verwendet die Metasprache DTD (Document Type Definition) zur Definition konkreter Sprachen
- XML (Extensible Markup Language) ist eine Untermenge von SGML
 - ⇒ z.B. mit spezieller Klammerstruktur
- HTML und XHTML sind konkrete Sprachen
 - ⇒ HTML und XHTML definieren ausgehend von SGML bzw. XML eine Sprache (Dokumenttyp) mit Tags, Attributen und einer Grammatik
 - ⇒ XHTML beinhaltet strenge Regeln und behandelt Verstöße als Fehler
 - ⇒ HTML lässt viele Freiheiten (ist weniger eindeutig) und interpretiert tolerant
- HTML5 wurde von SGML losgelöst und als eigenständige Sprache definiert
 - ⇒ für HTML5 gibt es eine HTML-und eine XHTML-Syntax ("XHTML5")



<http://jasonpriem.org/2010/04/markup-languages-whos-who>

Bemerkungen zum Werdegang

- Die Entwicklung von HTML war ein Trauerspiel
 - ⇒ Anstatt ein ausgereiftes Dokument-Format um URLs zu erweitern wurde HTML seit 1992 jahrelang "neu" erfunden
 - ⇒ Der Kampf um Browser-Marktanteile führte zu proprietären HTML-Varianten
 - ⇒ der Standardisierungsprozess hinkte immer hinter dem Entwicklungsstand der Browser her
 - ⇒ Erst mit HTML4 gab es 1998 einen angemessenen Standard
 - ⇒ HTML5 wurde 10 (!) Jahre später (2009) als Working Draft veröffentlicht
 - ⇒ Die Browser halten bis heute keinen Standard vollständig ein und beinhalten Bugs – oder interpretieren den Standard speziell
- Leidtragende sind die Nutzer
 - ⇒ Webentwickler können ihre Ziele nur schwer realisieren
 - ⇒ Surfer leiden unbewusst (wissen nicht, wie es gemeint war)

Was ist eigentlich HTML5?



- Alle reden von HTML5 – und alle meinen unterschiedliche Dinge
 - ⇒ Das WWW-Konsortium "W3C" (<http://www.w3.org>) definiert traditionell den Standard und achtet auf eine saubere Darstellung
 - Üblicherweise wird der Standard in einem langen Prozess versioniert
 - Der Standard liegt weit hinter der Realität im Web zurück
 - ⇒ "WHATWG" (Web Hypertext Application Technology Working Group) beschreibt als "Living Standard" die neuesten Entwicklungen
 - Änderungen sind in der Webwelt normal und sollen in einem schnellen Prozess in den Living Standard von HTML5 aufgenommen werden
 - Die Änderungen sind fester Bestandteil von HTML5
 - HTML5 ist keine Version, sondern ein eigenständiger Name (deshalb fehlt auch das Leerzeichen vor der 5)
 - ⇒ In der Umgangssprache steht HTML5 für das interaktive Internet mit ansprechendem Design (Web 2.0)
 - Das beinhaltet nicht nur HTML5, sondern eine Mischung diverser Webtechniken wie CSS, JavaScript, Frameworks, APIs



2.1 HTML

Was bringen HTML5 & Co.?

- HTML5 & Co. bringen viele Neuerungen. Einige davon sind
 - ⇒ Ausnutzung von leistungsstarken Clients
 - Aufwändige Grafiken (Canvas)
 - Unterstützung von Threads (Workers in Javascript)
 - Datenspeicherung (MicroData)
 - ⇒ Unterstützung von unterschiedlichen Clients
 - Erkennung der Auflösung (CSS3 Media Queries)
 - Offlinezugriffe und Lokalisierung
 - ⇒ Modernisierung des alten Standards
 - Eingabeelemente mit Überprüfungen (z.B. Email-Felder)
 - Audio und Video jetzt integriert
 - Neue Tags (z.B. Header, Footer, Navigationsbereich)
 - Schickeres Design (z.B. Schatten und runde Ecken in CSS3)

HTML5 ist das, was man im Web heute braucht!



2.1 HTML

HTML5: Pro und Contra

■ Contra

- ⇒ W3C und WHATWG propagieren unterschiedliche Standards
- ⇒ Neue Tags und Attribute werden von vielen Browsern nicht erkannt (das ist nicht wirklich neu)

■ Pro

- ⇒ HTML5 wurde (bzw. wird) abwärtskompatibel entwickelt und selbst alte Browser reagieren gnädig auf unbekannte Inhalte
- ⇒ Die neue Funktionalität mit HTML4 zu implementieren ist kaum machbar
- ⇒ Es gibt diverse Webseiten, die zeigen wie weit die neuen Features in den verschiedenen Browsern umgesetzt sind (z.B. <http://canluse.com>)
- ⇒ Es gibt diverse JavaScript-Frameworks, die dabei helfen alte Browser zu unterstützen (z.B. <http://modernizr.com>)

Wir verwenden HTML5 !

HTML5 und ordentlicher Code

- HTML5 wurde abwärtskompatibel und browserfreundlich definiert
 - ⇒ für viele Sprachkonstrukte, die unter HTML4 einen Fehler erzeugten, ist jetzt das Browserverhalten definiert – und es sind keine Fehler mehr
 - viele fehlende Tags werden automatisch erkannt
 - schließende Tags können oft weggelassen werden
 - Groß/Kleinschreibung der Tags ist nicht festgelegt usw.
 - ⇒ Praktische Konsequenzen
 - Code der gegen viele Regeln eines sauberen Entwurfs verstößt, kann trotzdem HTML5-konform sein
 - um HTML5-Code auf einen ordentlichen Stil zu überprüfen, reicht der normale Konformitätscheck (HTML-Validator) nicht mehr aus
 - ⇒ Lösung
 - HTML5 sollte in professionellen Projekten zusätzlich mit einem statischen Code-Analyse-Tool auf die Einhaltung von Implementierungsrichtlinien geprüft werden (vgl. Lint für C++)

Implementierungsrichtlinien in dieser Veranstaltung

■ Implementierungsrichtlinien für EWA

- ⇒ Kein Layout / Design in HTML sondern konsequenter Einsatz von CSS
- ⇒ Einrückungen wie in normalen Programmiersprachen
- ⇒ Weitere Regeln, die HTML5 einschränken, werden auf den folgenden Folien mit diesem Symbol markiert



■ Die Einschränkungen sind gering und fordern keine große Umstellung

- ⇒ Die Syntax für ein paar HTML-Konstrukte ist etwas strenger
- ⇒ Der HTML-Code wird aber definitiv besser lesbar

Ordentlicher HTML5-Code ist die Zielsetzung in dieser Veranstaltung!