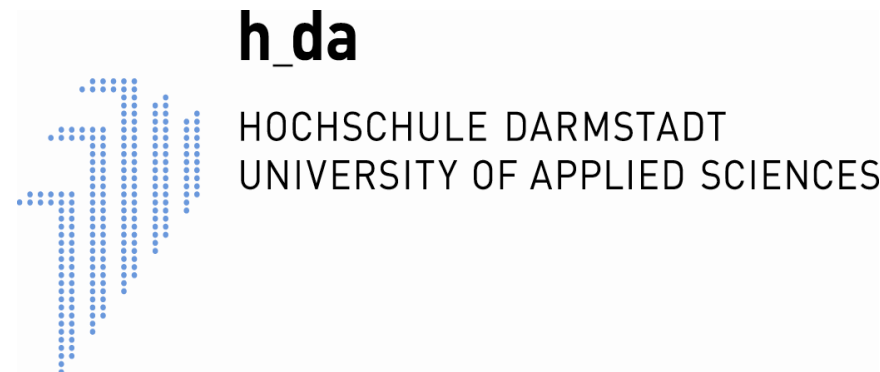


# Entwicklung webbasierter Anwendungen

Prof. Dr. Ralf Hahn im SS 2006

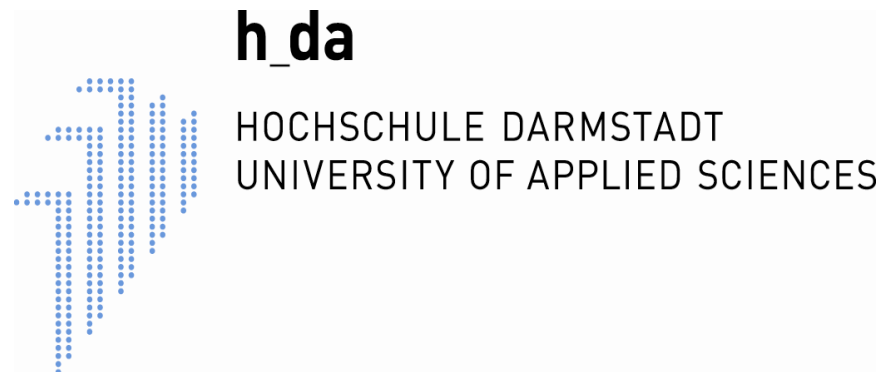


Quellenhinweis:

Viele Folien dieser Vorlesung entstammen der gleichnamigen Vorlesung von Prof. B. Kreling

# Entwicklung webbasierter Anwendungen

## 1. Kapitel: Einleitung



Quellenhinweis:

Viele Folien dieser Vorlesung entstammen der gleichnamigen Vorlesung von Prof. B. Kreling

## Zielsetzung

### ■ aus der Modulbeschreibung:

- ⇒ Die Studierenden sollen Methoden und Techniken zur Entwicklung von webbasierten Anwendungen kennen, verstehen und anwenden können. Dies gilt insbesondere auch für datenbankbasierte Web-Anwendungen.
- ⇒ Sie sollen die Grundlagen und die Handhabung eines aktuellen Entwicklungswerkzeugs für animationsorientierte webbasierte Anwendungen kennen.






### ■ Konkret:

Nach der Veranstaltung...

- ⇒ kennen Sie den Sinn, Zweck und die Grenzen der verschiedenen Techniken
- ⇒ verstehen Sie das Zusammenspiel der verschiedenen Techniken
- ⇒ kennen Sie die wesentlichen Standards
- ⇒ sind Sie in der Lage, komplexe und standardkonforme Webseiten zu erstellen

# 1. Einleitung






## Aufgabe im Praktikum: Pizzaservice

Kunde (Bestellung)		Pizzabäcker (bestellte Pizze)																												
 Margherita 4.00 €  Salami 4.50 €  Hawaii 5.50 €  Tonno 5.00 €  Speziale 7.00 €	<b>Warenkorb</b> <div style="border: 1px solid gray; padding: 5px; min-height: 100px;">           Salami            Salami            Hawaii         </div> <p style="text-align: right;">14.50 €</p> <div style="text-align: center;"> <input type="button" value="Bestellen"/>  <input type="button" value="Alle Löschen"/>  <input type="button" value="Auswahl Löschen"/> </div>	bestellt im Ofen Hawaii <input checked="" type="radio"/> <input type="radio"/> Salami <input type="radio"/> <input checked="" type="radio"/> Margherita <input type="radio"/> <input checked="" type="radio"/> Margherita <input type="radio"/> <input checked="" type="radio"/>																												
Kunde (Lieferstand)		Fahrer (fertige Pizze)																												
	im Ofen fertig unterwegs Margherita <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Salami <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> Tonno <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Hawaii <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/>	<table border="0"> <thead> <tr> <th></th> <th>fertig</th> <th>unterwegs</th> </tr> </thead> <tbody> <tr> <td>Salami</td> <td><input checked="" type="radio"/></td> <td><input type="radio"/></td> </tr> <tr> <td>Margherita</td> <td><input checked="" type="radio"/></td> <td><input type="radio"/></td> </tr> <tr> <td><b>FHD</b></td> <td></td> <td><b>8.5 €</b></td> </tr> <tr> <td>Tonno</td> <td><input checked="" type="radio"/></td> <td><input type="radio"/></td> </tr> <tr> <td>Tonno</td> <td><input checked="" type="radio"/></td> <td><input type="radio"/></td> </tr> <tr> <td><b>Müller, Freßgasse 11</b></td> <td></td> <td><b>10 €</b></td> </tr> <tr> <td>Margherita</td> <td><input type="radio"/></td> <td><input checked="" type="radio"/></td> </tr> <tr> <td><b>Hahn, Schöfferstr.8b</b></td> <td></td> <td><b>4 €</b></td> </tr> </tbody> </table>		fertig	unterwegs	Salami	<input checked="" type="radio"/>	<input type="radio"/>	Margherita	<input checked="" type="radio"/>	<input type="radio"/>	<b>FHD</b>		<b>8.5 €</b>	Tonno	<input checked="" type="radio"/>	<input type="radio"/>	Tonno	<input checked="" type="radio"/>	<input type="radio"/>	<b>Müller, Freßgasse 11</b>		<b>10 €</b>	Margherita	<input type="radio"/>	<input checked="" type="radio"/>	<b>Hahn, Schöfferstr.8b</b>		<b>4 €</b>	
	fertig	unterwegs																												
Salami	<input checked="" type="radio"/>	<input type="radio"/>																												
Margherita	<input checked="" type="radio"/>	<input type="radio"/>																												
<b>FHD</b>		<b>8.5 €</b>																												
Tonno	<input checked="" type="radio"/>	<input type="radio"/>																												
Tonno	<input checked="" type="radio"/>	<input type="radio"/>																												
<b>Müller, Freßgasse 11</b>		<b>10 €</b>																												
Margherita	<input type="radio"/>	<input checked="" type="radio"/>																												
<b>Hahn, Schöfferstr.8b</b>		<b>4 €</b>																												

## 1. Einleitung

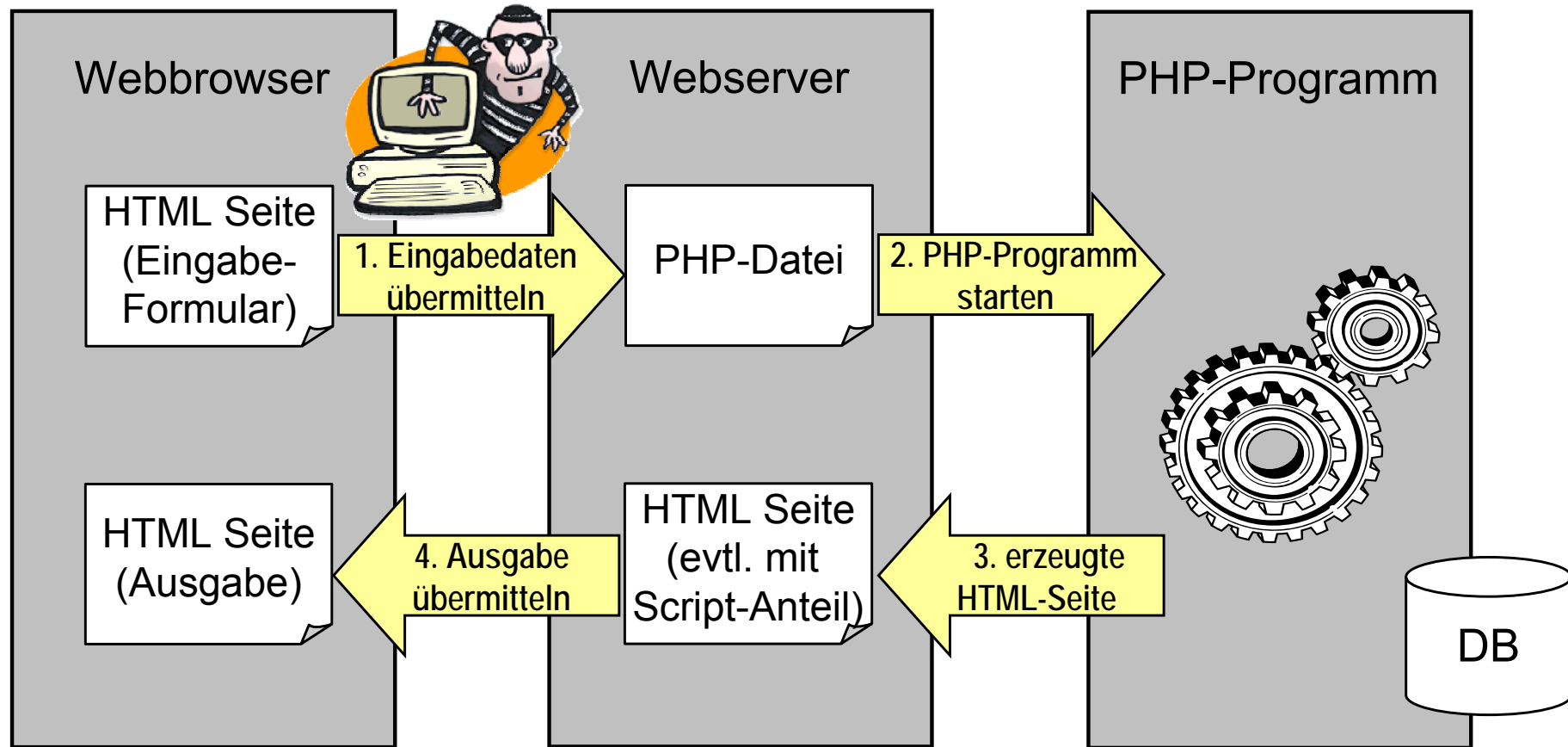
# Konkrete Inhalte des Veranstaltung

- (Entwurf)
- HTML Grundlagen
- CSS und dynamisches Layout
- Formulare und ECMAScript, DOM
- Server Konfiguration (Apache), CGI
- PHP mit Datenbankbindung (MySQL)
- Animation (Macromedia Flash)
- Sicherheit im Internet

Kunde (Bestellung)		Pizzabäcker (bestellte Pizze)																												
 Margherita 4.00 €	<b>Warenkorb</b> <input type="text" value="Salami"/> 14.50 € <input type="button" value="Bestellen"/> <input type="button" value="Alle Löschen"/> <input type="button" value="Auswahl Löschen"/>	bestellt im Ofen																												
 Salami 4.50 €		Hawaii <input checked="" type="radio"/> <input type="radio"/>																												
 Hawaii 5.50 €		Salami <input type="radio"/> <input checked="" type="radio"/>																												
 Tonno 5.00 €		Margherita <input type="radio"/> <input checked="" type="radio"/>																												
 Speziale 7.00 €		Margherita <input type="radio"/> <input checked="" type="radio"/>																												
Kunde (Lieferstand)		Fahrer (fertige Pizze)																												
	<table><thead><tr><th></th><th>im Ofen</th><th>fertig</th><th>unterwegs</th></tr></thead><tbody><tr><td>Margherita</td><td><input type="radio"/></td><td><input checked="" type="radio"/></td><td><input type="radio"/></td></tr><tr><td>Salami</td><td><input checked="" type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td></tr><tr><td>Tonno</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input checked="" type="radio"/></td></tr><tr><td>Hawaii</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input checked="" type="radio"/></td></tr></tbody></table>		im Ofen	fertig	unterwegs	Margherita	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Salami	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tonno	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Hawaii	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>									
	im Ofen	fertig	unterwegs																											
Margherita	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>																											
Salami	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>																											
Tonno	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>																											
Hawaii	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>																											
			<table><thead><tr><th></th><th>fertig</th><th>unterwegs</th></tr></thead><tbody><tr><td>Salami</td><td><input checked="" type="radio"/></td><td><input type="radio"/></td></tr><tr><td>Margherita</td><td><input checked="" type="radio"/></td><td><input type="radio"/></td></tr><tr><td><b>FHD</b></td><td></td><td><b>8.5 €</b></td></tr><tr><td>Tonno</td><td><input checked="" type="radio"/></td><td><input type="radio"/></td></tr><tr><td>Tonno</td><td><input checked="" type="radio"/></td><td><input type="radio"/></td></tr><tr><td><b>Müller, Freßgasse 11</b></td><td></td><td><b>10 €</b></td></tr><tr><td>Margherita</td><td><input type="radio"/></td><td><input checked="" type="radio"/></td></tr><tr><td><b>Hahn, Schöffstr.8b</b></td><td></td><td><b>4 €</b></td></tr></tbody></table>		fertig	unterwegs	Salami	<input checked="" type="radio"/>	<input type="radio"/>	Margherita	<input checked="" type="radio"/>	<input type="radio"/>	<b>FHD</b>		<b>8.5 €</b>	Tonno	<input checked="" type="radio"/>	<input type="radio"/>	Tonno	<input checked="" type="radio"/>	<input type="radio"/>	<b>Müller, Freßgasse 11</b>		<b>10 €</b>	Margherita	<input type="radio"/>	<input checked="" type="radio"/>	<b>Hahn, Schöffstr.8b</b>		<b>4 €</b>
	fertig	unterwegs																												
Salami	<input checked="" type="radio"/>	<input type="radio"/>																												
Margherita	<input checked="" type="radio"/>	<input type="radio"/>																												
<b>FHD</b>		<b>8.5 €</b>																												
Tonno	<input checked="" type="radio"/>	<input type="radio"/>																												
Tonno	<input checked="" type="radio"/>	<input type="radio"/>																												
<b>Müller, Freßgasse 11</b>		<b>10 €</b>																												
Margherita	<input type="radio"/>	<input checked="" type="radio"/>																												
<b>Hahn, Schöffstr.8b</b>		<b>4 €</b>																												

# 1. Einleitung

## Einsatz der Technologien im Zusammenhang



- HTML
- CSS
- ECMA-Script
- DOM
- Animation

- HTTP
- Server-Konfiguration

- CGI
- PHP
- MySQL

# Problem 1: Transferraten

<b>Modem analog</b>	<b>56 kBaud</b>	<b>5,6 kByte/s</b>
<b>ISDN</b>	<b>64 kb/s</b>	<b>8 kB/s</b>
<b>DSL</b>	<b>768 kb/s</b>	<b>96 kB/s</b>
<b>CD-ROM (1x)</b>	<b>1,2 Mb/s</b>	<b>150 kB/s</b>
<b>CD-ROM (12x)</b>	<b>14,4 Mb/s</b>	<b>1800 kB/s</b>
<b>UMTS</b>	<b>1,2 Mb/s</b>	<b>150 kB/s</b>
<b>Ethernet LAN (alt)</b>	<b>10 Mb/s</b>	<b>1.220 kB/s</b>

galt mal als "enabling technology" für MM

MPEG1 oder einfaches AVI-Video

⇒ man muss immer noch mit Datenvolumen geizen !

## Problem 2: Plattformabhängigkeit

### ■ "Plattform" bisher: Betriebssystem

- ⇒ oft nur für bestimmte Rechnertypen verfügbar
- ⇒ Unix-Varianten (Workstation), Windows (PC), MacOS (Mac), MVS (Mainframe)

### ■ "Plattform" im Web: Browser + Version

- ⇒ Standard durch inkompatible Spracherweiterungen ausgehebelt
- ⇒ manche Plug-Ins nur für bestimmte Betriebssysteme verfügbar
  - betrifft insbesondere MM-Dienste, weil diese wiederum Betriebssystemdienste nutzen (vgl. Video)

### ■ kein technisches Problem, sondern marktpolitisches

lediglich eine  
Verlagerung der  
Abhängigkeit

## Problem 3: what you see is NOT what you get !

### ■ Darstellung erfolgt über HTML

- ⇒ HTML ist eine Markup Language
- ⇒ WYSIWYG ist per Prinzip nicht möglich
- ⇒ HTML Editor zeigt allenfalls ungefähr das Ergebnis
- ⇒ sorgfältige Vorschau notwendig mit verschiedenen Browsern / Bildschirmauflösungen / Fenstergrößen

wieso hat man  
das Problem  
z.B. in  
PowerPoint  
nicht ?

### ■ Surfer-freundliche Darstellung

- ⇒ nicht unterstützte HTML-Anweisungen werden nicht gemeldet, sondern ignoriert
- ⇒ Darstellung so gut es eben geht
- ⇒ unangenehm für Webdesigner: visuelle Kontrolle erforderlich
- ⇒ unbedingt Tool (z.B. HTML Validator) verwenden

## Literatur und Software

### Literatur

- Stefan Münz: SELFHTML  
<http://www.selfhtml.org>
- Damir Enseleit: SELFPHP  
<http://www.selfphp.info>
- Ed Tittel, Mary Burmeister:  
*HTML 4 für Dummies*, Wiley-Vch 2005
- Huseby: *Sicherheitsrisiko Webanwendung*, dpunkt.verlag 2004

### Standards

- HTML-, CSS-, DOM-Standard  
und HTML/CSS-Validator  
<http://www.w3.org/>
- ECMAScript (ECMA-262)  
<http://www.ecma-international.org/>

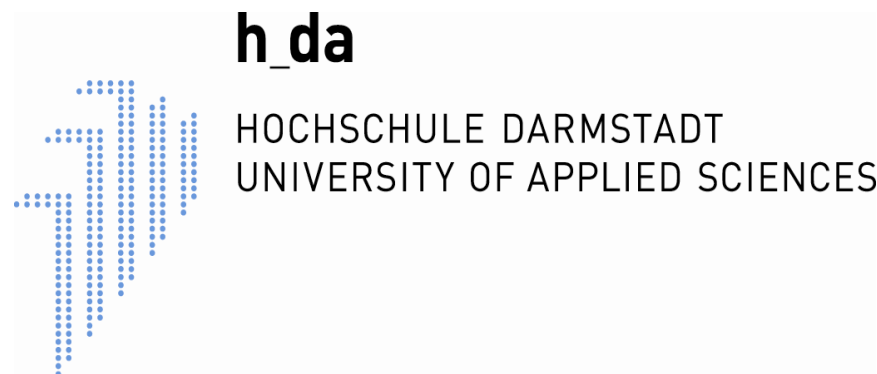
### Freie Software, Dokus, Tutorials

- HTML Editor + Validator  
<http://www.meybohm.de/>
- HTML Editor mit Preview  
<http://www.evrsoft.com/1stpage2.shtml>
- XAMPP (Webserver, MySQL, PHP)  
<http://www.apachefriends.org/de/xampp.html>

# Entwicklung webbasierter Anwendungen

## 2. Kapitel:

### Softwaretechnik für webbasierte Anwendungen



Quellenhinweis:

Viele Folien dieser Vorlesung entstammen der gleichnamigen Vorlesung von Prof. B. Kreling

## Motivation

- auch Multimedia-Anwendungen und WebSites sind Softwaresysteme !
  - ⇒ es gilt weiterhin alles, was man über Softwaretechnik, Software Ergonomie und GUIs gelernt hat

- nicht vor lauter

Design



Grafik, Animation, Gimmicks

den

Entwurf



Analyse, Architektur, Datenorganisation

vergessen

- Die Programmiersprachen und die Aufgabe verleiten zum Hacken !
- Ein Konzept hat aber noch nie geschadet !

## Anforderungsanalyse

### ■ Zweck des Produkts bestimmen

- ⇒ Was wollen Benutzer mit der Anwendung erreichen?  
"sich informieren" ist zu wenig!
- ⇒ Produktkatalog, Selbstlernmedium, Spiel, Werbung,...?
- ⇒ Fülle von Informationen darstellen und dennoch leichte Orientierung?

### ■ Ermittlung der Zielgruppe

- ⇒ Alter, Geschlecht, Sprache, Ausstattung, Ausbildung, PC-Erfahrung, Internet-Zugang, Benutzungsfrequenz

### ■ Verkaufsargument

- ⇒ Nutzen, Mehrwert ggü. Konkurrenz (z.B. Printmedien)

### ■ Erscheinungsbild

- ⇒ reine Information oder auch Darstellung

### ■ Zielmedium

- ⇒ CD, Internet, Intranet, Fernseher (Set-Top-Box)

## Randbedingungen (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, Couch Potatoes
- 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

## Design: Strukturierungsmittel

### ■ Allgemein

- ⇒ Dokumentenstruktur: Daten, Struktur, Format
- ⇒ Navigationsstruktur: Netz oder Baum oder Folge oder ...

### ■ Suchhilfsmittel

- ⇒ Inhaltsverzeichnis, Index, Volltextsuche, History, Lesezeichen

### ■ Verwenden von Metaphern hilft oft

- ⇒ Museum mit Ausstellungen
- ⇒ Fernseher mit Kanälen
- ⇒ Dateisystem mit Ordnern und Dokumenten

## Design: Interaktionselemente

### ■ als Interaktionselement erkennbar

Minimalforderung

- ⇒ d.h. unterscheidbar von passiven Elementen
- ⇒ Button: durch Gestalt
- ⇒ Hotword: durch Farbe / Unterstreichung

### ■ Zustandsdarstellung

vermeidet nutzlose Betätigung

- ⇒ normal, gesperrt, evtl. aktiviert / gedrückt
- ⇒ Button: graphische Variante (z.B. gesperrt: Schrift grau)
- ⇒ Hotword, sensitiver Bereich: nicht üblich

### ■ Erkennungsaufwand

Adventure oder Ergonomie ?

- ⇒ unmittelbar visuell (Button, Hotword): gering
- ⇒ Mauszeiger ändert sich beim Abtasten: hoch  
(nur akzeptabel als Ergänzung)

## Mensch-Maschine-Schnittstelle

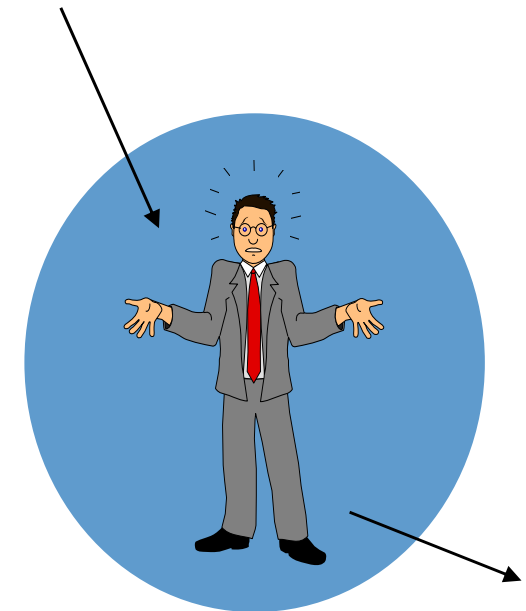
- Der Systemzustand muss auf einen Blick erkennbar sein
  - ⇒ Grund-Forderung aus der Software-Ergonomie (ohne ihn dabei zu verändern!)
- Wo bin ich ?
  - ⇒ momentaner Aufenthaltsort im System
- Was kann ich hier tun ?
  - ⇒ zur Verfügung stehende Operationen
- Wie kam ich hierher ?
  - ⇒ Vorgeschichte,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

Ort

Modus

Weg

Weg



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

## Layout: Komponenten einer Bildschirmseite

### ■ Kopfzeilenkomponente

wo bin ich?

⇒ dient der Orientierung des Benutzers

### ■ Präsentationskomponente

was kann ich hier tun?

⇒ Darstellung der Informationsgehalts

⇒ Bühne für Animationen, Simulationen, Videos

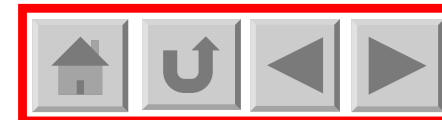
### ■ Navigationskomponente (Interaktions-) wohin kann ich gehen?

⇒ dient der Steuerung durch den Benutzer

### ■ Hintergrund

⇒ passives Designelement

⇒ unverändert über mehrere Bildschirmseiten



# Layoutbeispiel: Lernprogramm "ToolBook"

mustergültig

**Multimedia ToolBook erlernen**

## Objekte und ihre Eigenschaften 46 -> 2.3.3 (12)

**Benutzerdefinierte Objekte**  
 ▶ Schaltflächen

Schaltflächen sind folgende Objekttypen:  
 Druckschalter, Optionsfelder, Kontrollkästchen und Aufschriftschaltflächen. Schaltflächen werden oft verwendet, um Ereignisse auszulösen oder Eigenschaften festzulegen.

Jede Schaltfläche hat vier Arbeitszustände:

Arbeitszustand	Eigenschaften:
Normal	---
Invertiert	invert
Gesperrt	not enabled
Aktiviert	checked

Sie können die verschiedenen Arbeitszustände einer Schaltfläche mit verschiedenen [Grafikressourcen](#) verbinden.

**Im normalen Zustand**  
 ▶ Druckschalter     Kontrollkästchen     Optionsfeld

**Im invertierten Zustand**  
 ▶ Druckschalter     Kontrollkästchen     Optionsfeld

**Im gesperrten Zustand**  
 ▶ Druckschalter

**Im aktivierten Zustand**  
 Kontrollkästchen     Optionsfeld

Beenden    Optionen    Plan    Abgeschlossene Lektionen im Abschnitt

K

P

N

# Test

## ■ Ganz normale Tests !

- ⇒ Funktionalität
- ⇒ nicht-funktionale Anforderungen
- ⇒ Ausnahmefälle

## ■ Besonders wichtig

- ⇒ Verständlichkeit und Bedienbarkeit
  - mit dem Auftraggeber bzw. künftigen Benutzern
- ⇒ Portabilität
  - Browser, Plugins, CPU, Bildschirme, Datenverbindung
- ⇒ Performanz
  - verschiedene Browser, Ausgabegeräte
  - geringe Transferrate
  - hohe Benutzerlast

## Vorgehensmodell

### ■ Besonderheiten bei Multimedia- und Webapplikationen

- ⇒ Usability ist extrem wichtig
- ⇒ hoher Gestaltungsanteil

### ■ Vorgehen

- ⇒ Wasserfall ist in der Regel ungeeignet
- ⇒ besser iterative Vorgehensweise
- ⇒ evtl. Prototypen einsetzen
- ⇒ mehrere Durchgänge mit zunehmender Verfeinerung und Präzisierung

### ■ Zwischenergebnisse

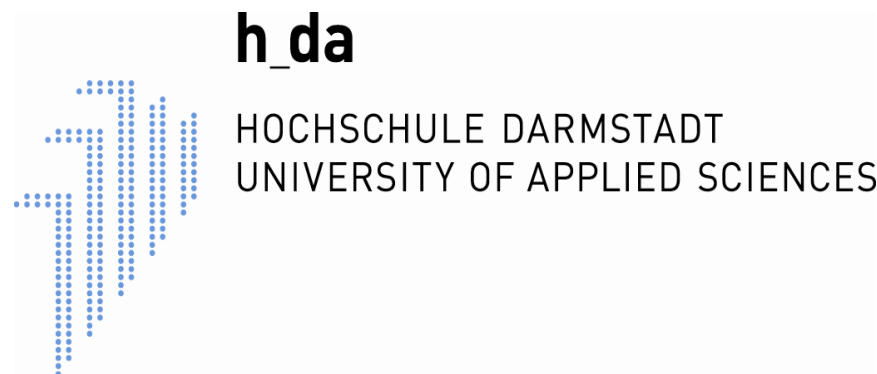
- ⇒ mit Auftraggeber und künftigen Benutzern abstimmen

## Arbeitsergebnisse (Meilensteine)

- Visionsdokument mit Sinn und Zweck der Website
- Übersicht der Funktionalität (Use Case Diagramm)
- Beschreibung der nicht-funktionalen Anforderungen
- Navigationsübersicht (Zustandsdiagramm)
- Skizzen der Seite(n)
- Modellierung der SW-Struktur
  - ⇒ Aktivitätsdiagramm
  - ⇒ Klassendiagramm
  - ⇒ Sequenzdiagramm (Kollaborationsdiagramm)
    - Zusammenspiel von HTML-Seiten (Client) und PHP-Seiten (Server)
- Testplan

# Entwicklung webbasierter Anwendungen

## 3. Kapitel: HTML



Quellenhinweis:

Viele Folien dieser Vorlesung entstammen der gleichnamigen Vorlesung von Prof. B. Kreling

## Was ist HTML ?

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

### ■ HyperText Markup Language

⇒ definiert mit SGML

(Standard Generalized Markup Language, ISO-Norm 8879)

### ■ Markup Language: Auszeichnungssprache

⇒ markiert und attributiert Bestandteile eines Dokuments

⇒ Browser setzen Auszeichnung in visuelle Darstellung um

### ■ Hyper Text: Verweise auf andere Dokumente

⇒ Hyperlinks zu anderen Stellen im eigenen Projekt oder zu beliebigen anderen Dokumenten im Web

### ■ Text-Dateien (kein Binärformat)

⇒ mit jedem Texteditor zu bearbeiten

⇒ leicht zu generieren und zu debuggen

## Ursprung: Hyper Text Markup Language

- semantisch gegliederter Fließtext
  - ⇒ Layout bewusst nicht definiert ⇒ inhaltsorientiert
  - ⇒ komfortable Querverweise durch Hyperlinks  
URL Uniform Resource Locator
- Tabellen und Formulare
- Einbetten von Bildern
  - ⇒ Audio oder Video nicht vorgesehen
- Layout war Sache des Browsers und des Surfers
  - ⇒ Schriftart und -größe wählbar
  - ⇒ Zeilenumbruch abhängig von Größe des Browserfensters

echte Innovation  
nach ftp, telnet, ...

## Entwicklungsgeschichte HTML

WWW Konsortium  
<http://www.w3.org>

### ■ Standardisierungsgremium W3C

⇒ ursprünglich Wissenschaftler, jetzt auch Industrievertreter

### ■ der Standardisierungsprozess hinkte immer hinter dem Entwicklungsstand kommerzieller Browser her

⇒ bewusst inkompatible Erweiterungen der Hersteller

⇒ Browser hielten keinen Standard vollständig ein; viele Bugs

### ■ klares Marketingziel

⇒ "best viewed with ..."

### ■ Webdesigner sind Leidtragende

⇒ gestalterische Ziele schwer zu realisieren

⇒ Surfer leiden unbewusst (wissen nicht, wie's gemeint war)

#### HTML Standardisierung durch W3C

1.0		
2.0	Nov 95	
3.2	Mai 96	Tabellen, phys. Format
4.0	Jan 98	Frames, CSS, Skript

## Bemerkungen zum Werdegang

- nach ca. 5 Jahren der Evolution beinhaltet HTML 4.0 endlich die Features, die in lokalen Anwendungen längst Stand der Technik waren
  - ⇒ Maßstab: Desktop Publishing (DTP) und Autorensysteme
  - ⇒ wichtiger Schritt: Cascading Style Sheets CSS
  
- politischer Aspekt war lange viel interessanter als technischer
  - ⇒ Browser-Wettkampf
  - ⇒ man hätte auch gleich ein ausgereiftes Dokument-Format standardisieren und um URLs erweitern können ...
  
- langer Umweg über HTML "Standardisierung"
  - ⇒ Versuch der Browser-Hersteller, proprietäre HTML Varianten zu etablieren
  - ⇒ planlose Erweiterungen
  - ⇒ Tag-Bastelei statt systematischen Lösung

## Zielrichtung in dieser Veranstaltung

- HTML 4.0 bereinigt Irrwege der Vergangenheit
  - ⇒ Variante "strict" (Alternativen "transitional" und "frameset")
  - ⇒ etliche Tags und Attribute sind "deprecated" (missbilligt)
- XHTML definiert HTML konform zu XML
  - ⇒ in der Praxis geringfügige Modifikationen ⇒ zukunftsorientiert
- Orientierung an Standards, nicht an Browsern
  
- praktische Konsequenzen
  - ⇒ Verzicht auf physische Formatierung in HTML, stattdessen konsequenter Einsatz von CSS
  - ⇒ Tags immer in Kleinbuchstaben
  - ⇒ zu jedem öffnenden Tag ein schliessendes
  - ⇒ alle Attribute in Anführungszeichen