Kapitel 8: Cascading Style Sheets

Literatur:

- Erik Wilde: World Wide Web — Technische Grundlagen. Springer, 1999, ISBN 3-540-64700-7, 641 Seiten.
- Eric Ladd, Jim O'Donnell, et al.: Using HTML 4, XML, and Java 1.2, Platinum Edition. QUE, 1999, ISBN 0-7897-1759-X, 1282 pages.
- W3C: Web Style Sheets Home Page. `http://www.w3.org/Style/`
- W3C: Cascading Style Sheets Home Page. `http://www.w3.org/Style/CSS/`
- Håkon Wium Lie, Bert Bos: Cascading Style Sheets, level 1. W3C Recommendation.
  `http://www.w3.org/TR/REC-CSS1-961217` (CSS1, Version vom 17.12.1996)
  `http://www.w3.org/TR/REC-CSS1` (CSS1, Neueste Version, 11.01.1999)
- Bert Bos, Håkon Lie, Chris Lilley, Ian Jacobs: Cascading Style Sheets, level 2. W3C Recommendation.
  `http://www.w3.org/TR/REC-CSS2/`
- Dave Raggett, Arnaud Le Hors, Ian Jacobs (Eds.): HTML 4.01 Specification. W3C, 24.12.1999.
  `http://www.w3.org/TR/html4/`
- Stefan Münz: HTML-Dateien selbst erstellen — SELFHTML. `http://www.netzwelt.com/selfhtml/`
- Ian Graham: Introduction to HTML.
  `http://www.utoronto.ca/webdocs/HTMLdocs/NewHTML/htmlindex.html`

Lernziele

- Verständnis für den Nutzen von Style Sheets.
- Eigene Entwicklung von Style Sheets, zumindest in relativ einfachen Fällen.
- Übersicht über Literatur/Quellen für weitere Informationen.

Inhalt

Was sind Style Sheets (1)

- Man unterscheidet zwei Arten von Markup (s.o.):
  - Inhalts-orientiertes (logisches, deskriptives) Markup und
  - Darstellungs-orientiertes (physisches, prozedurales) Markup.
- WWW-Dokumente können auf verschiedene Arten verwendet werden:
  - Darstellung im Browser-Fenster.
  - Ausdruck auf Papier.
  - Ausgabe in Blindenschrift, Sprachausgabe.
  - Eintragung in Suchmaschinen.
  - Verarbeitung durch Programme (Agenten, Shopbots, etc.)

Was sind Style Sheets (2)

- HTML hatte anfangs vor allem Inhalts-orientierte Elemente.
- Dann führten aber die Browser-Hersteller (zuerst vor allem Netscape) viele neue Elemente und Attribute ein, die das Aussehen der Seiten im Browser-Fenster beschreiben.

  Dies hatte auch den Zweck, Benutzer an einen Browser zu binden. Auf den Seiten stand oft „`Best viewed with ...`“.
- So wurde HTML immer umfangreicher, undurchschaubarer, und schwieriger für andere Medien zu verarbeiten.

  HTML war so auch einer schnellen Änderung unterworfen.
- Deswegen wurde bei HTML 4 eine Trennung eingeführt: HTML sollte nur den Inhalt beschreiben, das Aussehen sollte mit Hilfe von Stylesheets festgelegt werden.

Was sind Style Sheets (3)

- Die Angaben zum Aussehen sind häufig willkürlich (Geschmackssache). Auch deswegen ist eine Trennung vom eigentlichen Inhalt wünschenswert.
- Für Anwendungen wie Suchmaschinen sind Angaben zum Aussehen irrelevant.
- Für andere Medien (Ausdrucke, Sprachausgaben) sind andere Festlegungen zum Aussehen notwendig.
- Es kann zu einer HTML-Datei verschiedene Stylesheets geben.
- Man kann auch umgekehrt in vielen HTML-Dateien das gleiche Stylesheet verwenden (einheitliches Aussehen).

Was sind Style Sheets (4)

- HTML ist nicht an eine bestimmte Stylesheet-Sprache gebunden.
- Man sollte deklarieren, daß man „Cascading Stylesheets“ (CSS) verwendet.
- Das Wort „Cascading“ bezieht sich auf die Möglichkeit, mehrere Stylesheets zu vermischen (d.h. Angaben verschiedener Prioritäten zu haben).

  Zum Beispiel könnte die Firma ein Stylesheet für ihre „Corporate Identity“ vorgeben, das eventuell für jede Abteilung leicht modifiziert werden kann, anschließend für jedes Dokument, und dann kann es noch Ausnahmen in den einzelnen Elementen im Text geben.

Was sind Style Sheets (5)

- Mit CSS1 kann man z.B. definieren:
  - Schriftart, Schriftgröße
  - Farben
  - Einrückungen, horizontaler und vertikaler Leerplatz
  - Begrenzungslinien, Ränder
  - Ausrichtung, z.B. Zentrierung
- Insgesamt bekommt man mehr Möglichkeiten, HTML jemals an Darstellungs-orientierten Elementen und Attributen hatte.
- Theoretisch kann man mit Stylesheets auch das übliche Verhalten der Browser definieren („Default-Stylesheet“).

Geschichte

- Cascading Style Sheets, level 1 (CSS1) wurde vom W3C am 17.12.1996 verabschiedet.

  Eine überarbeitete Version des Standards wurde am 11.01.1999 veröffentlicht.
- Cascading Style Sheets, level 2 (CSS2) wurde am 12.05.1998 verabschiedet. Das W3C arbeitet an CSS3.
- Der Internet Explorer kennt einen Teil von CSS1 seit Version 3.0. In der aktuellen Version soll CSS1 vollständig implementiert sein, sowie einige Befehle von CSS2.
- Netscape 3.x verstand noch keine Style Sheets, aber Netscape 4.x versteht den größten Teil von CSS1.

Inhalt

Einbindung in HTML (1)

- Stylesheet-Angaben können ein drei Stellen stehen:
  - In einer vom HTML-Dokument getrennten Datei, auf die mit `LINK` verwiesen wird.

    Nachteil: Der Browser muß die Stylesheet-Information extra anfordern und kann den Text erst danach darstellen.
    Vorteil: Ein Stylesheet für mehrere Seiten möglich.
    Nur benötigte Stylesheets werden geladen.
  - Innerhalb des Elementes `STYLE` im Dokument-Kopf.

    Nachteil: Keine Mehrfachverwendung von Stylesheets.
  - Jeweils in dem betroffenen Element im Attribut `style`.

    Nachteil: Man kann keine alternativen Angaben machen.
    Inkonsistenzen durch sehr lokale Angaben.

Einbindung in HTML (2)

- Z.B.: `H1` Überschriften sollen in roter 14pt Schrift erscheinen.
- Man legt eine Datei, z.B. `my.css`, mit folgendem Inhalt an:

```
H1: { color: red; font-size: 14pt}
```

- Diese kann man nun mit einem `LINK`-Element referenzieren:

```
<LINK href="my.css"
      rel="stylesheet" type="text/css">
```

- Möchte man, daß das Stylesheet auch zum Ausdrucken verwendet wird, so setzte man im `LINK`-Element noch

```
media="screen,print"
```

  Der Default ist `media=screen`.

  Für weitere Ausgabemedien siehe Kapitel 5 unter `LINK`.

Einbindung in HTML (3)

- Die Information kann auch im Element STYLE stehen:

```
<STYLE type="text/css">
    H1: { color: red; font-size: 14pt}
</STYLE>
```

- Ältere Browser, die STYLE nicht kennen, nehmen an, daß hier bereits der BODY beginnt.
- Deswegen wird die Stylesheet-Information manchmal in einen HTML-Kommentar eingeschlossen:

```
<STYLE type="text/css"><!--
    H1: { color: red; font-size: 14pt}
--></STYLE>
```


Einbindung in HTML (4)

- Das Element STYLE ist folgendermaßen deklariert:

```
<!ELEMENT STYLE - -    CDATA >
<!ATTLIST STYLE %i18n;
                type   %ContentType; #REQUIRED
                media  %MediaDesc;   #IMPLIED
                title  %Text;        #IMPLIED >
```

- Da der Inhalt als CDATA deklariert ist, wird Markup nicht ausgewertet (insbesondere keine Kommentare gelöscht).
  Kommentare in Stylesheets schreibt man wie in C: /*...*/.
- Auch hier muß media gesetzt werden, wenn ein Stylesheet nicht oder nicht nur für den Bildschirm gedacht ist.

Einbindung in HTML (5)

- Alternativ kann man die Angabe auch direkt bei den betroffenen Überschriften machen:

```
<H1 style="color: red; font-size: 14pt">
```

- Das Attribut style ist Bestandteil von %attrs;.
  Es ist bei praktisch jedem Body-Element möglich.
- Der HTML Standard verlangt, daß die Stylesheet-Sprache immer deklariert ist.
- Während dies bei LINK und STYLE mit dem Attribut type geschieht, ist hier ein Meta-Tag nötig:

```
<META http-equiv="Content-Style-Type"
      content="text/css">
```


Mehrere Stylesheets (1)

- Es ist möglich, mehrere alternative Stylesheets im Dokument anzugeben, und dem Benutzer die Auswahl zu überlassen.
  Falls der Browser das unterstützt.
- Dazu hat man mehrere LINK-Elemente mit Verweisen auf Stylesheet-Dateien.
- Der vom Autor vorgeschlagene Default („preferred stylesheet“) verwendet rel=stylesheet, alle anderen verwenden

```
rel="alternate stylesheet"
```

- In diesem Fall muß das title-Attribut aller dieser LINK-Elemente gesetzt werden (der Browser kann dann diese Namen dem Benutzer zur Auswahl anbieten).

Mehrere Stylesheets (2)

- Wird für ein LINK-Element kein title angegeben, und ist rel=stylesheet, so zählt es als „persistent" und muß immer angewendet werden (ggf. zusätzlich zur Benutzerwahl).

  Natürlich nur, wenn die media-Auswahl paßt. Außerdem können des Browser erlauben, Stylesheets ganz abzuschalten.

- Zumindest bei CSS ist es möglich, Angaben aus mehreren Stylesheets zu mischen (z.B. mehrere passende LINK-Elemente oder LINK und zusätzlich STYLE etc.).
- Die Prioritätsregeln sind relativ kompliziert (s.u.), aber wenn es sonst keinen Unterschied gibt, gewinnt die spätere Angabe.

Inhalt

Auswahl von Elementen (1)

- Man kann im Stylesheet sich nicht nur auf Element-Typen beziehen, sondern auch
  - Werte der Attribute class, id:
    ```
    H1.myclass: { color: red; font-size: 14pt}
    #myid:      { color: red; font-size: 14pt}
    ```
  - LI-Elemente innerhalb eines UL-Elementes:
    ```
    UL LI: { color: red; font-size: 14pt}
    ```
  - Besuchte Links:
    ```
    A:visited: { color: red; font-size: 14pt}
    ```
  - Man kann auch mehrere Angaben zusammenfassen:
    ```
    H1, H2: { color: red; font-size: 14pt}
    ```


Inhalt

Stylesheet Angaben (1)

- Font properties: `font-family`, `font-style`, `font-variant`, `font-weight`, `font-size`, `font`.
- Color and Background Properties: `color`, `background-color`, `background-image`, `background-repeat`, `background-attachment`, `background-position`, `background`.
- Text properties: `word-spacing`, `letter-spacing`, `text-decoration`, `vertical-align`, `text-transform`, `text-align`, `text-indent`, `line-height`.

Stylesheet Angaben (2)

- Box Properties: `margin-top`, `margin-right`, `margin-bottom`, `margin-left`, `margin`, `padding-top`, `padding-right`, `padding-bottom`, `padding-left`, `padding`, `border-top-width`, `border-right-width`, `border-bottom-width`, `border-left-width`, `border-width`, `border-color`, `border-style`, `border-top`, `border-right`, `border-bottom`, `border-left`, `border`, `width`, `height`, `float`, `clear`.
- Classification Properties: `display`, `white-space`, `line-style-type`, `line-style-image`, `line-style-position`, `line-style`.