

Grundbegriffe der Informatik

Einheit 7: Dokumente

Thomas Worsch

Karlsruher Institut für Technologie, Fakultät für Informatik

Wintersemester 2010/2011

Dokumente

$\text{\LaTeX}$

XHTML

Eine Grenze unserer bisherigen Vorgehensweise

- ▶ im alltäglichen Leben vielerlei Inschriften:
Briefe, Kochrezepte, Zeitungsartikel, Vorlesungsskripte, Seiten im WWW, Emails, usw.
- ▶ oft drei verschiedene Aspekte unterscheidbar, die für den Leser eine Rolle spielen:
 - ▶ den **Inhalt** des Textes,
 - ▶ seine **Struktur** und
 - ▶ sein **Erscheinungsbild**, die (äußere) **Form**.

- ▶ den **Inhalt** des Textes,
- ▶ seine **Struktur** und
- ▶ sein **Erscheinungsbild**, die (äußere) **Form**.

- ▶ den **Inhalt** des Textes,
- ▶ seine **Struktur** und
- ▶ sein **Erscheinungsbild**, die (äußere) **Form**.

andere Form:

- ▶ den INHALT des Textes,
- ▶ seine STRUKTUR und
- ▶ sein ERSCHEINUNGSBILD, die (äußere) FORM.

- ▶ den **Inhalt** des Textes,
- ▶ seine **Struktur** und
- ▶ sein **Erscheinungsbild**, die (äußere) **Form**.

andere Struktur:

*[...] den **Inhalt** des Textes, seine **Struktur** und sein **Erscheinungsbild**, die (äußere) **Form**.*

- ▶ den **Inhalt** des Textes,
- ▶ seine **Struktur** und
- ▶ sein **Erscheinungsbild**, die (äußere) **Form**.

anderer Inhalt:

- ▶ **Balaenoptera musculus** (Blauwal),
- ▶ **Mesoplodon carlhubbsi** (Hubbs-Schnabelwal) und
- ▶ **Physeter macrocephalus** (Pottwal).

Wozu Inhalt, Struktur und Form?

- ▶ Inhalt üblicherweise für Autoren und Leser im Vordergrund
 - ▶ (Ausnahmen?)
- ▶ Struktur und Form
 - ▶ sollen den Leser beim Verstehen des Inhalts zu unterstützen.
- ▶ **Dokumente:** Texte mit Inhalt, Struktur und Form
 - ▶ z. B. Programme
- ▶ ein **Rat für Ihr weiteres Studium:**
 - ▶ viele Dokumente (PSE, Seminar, Bachelor-Arbeit, ...)
 - ▶ Finden geeigneter Struktur hilft auch eigenem Verständnis
 - ▶ *deswegen: früh damit beginnen, etwas aufzuschreiben*
(weil man dann über Struktur nachdenken muss)

Wozu Inhalt, Struktur und Form?

- ▶ Inhalt üblicherweise für Autoren und Leser im Vordergrund
 - ▶ (Ausnahmen?)
- ▶ Struktur und Form
 - ▶ sollen den Leser beim Verstehen des Inhalts zu unterstützen.
- ▶ **Dokumente:** Texte mit Inhalt, Struktur und Form
 - ▶ z. B. Programme
- ▶ ein Rat für Ihr weiteres Studium:
 - ▶ viele Dokumente (PSE, Seminar, Bachelor-Arbeit, ...)
 - ▶ Finden geeigneter Struktur hilft auch eigenem Verständnis
 - ▶ *deswegen: früh damit beginnen, etwas aufzuschreiben*
(weil man dann über Struktur nachdenken muss)

Wozu Inhalt, Struktur und Form?

- ▶ Inhalt üblicherweise für Autoren und Leser im Vordergrund
 - ▶ (Ausnahmen?)
- ▶ Struktur und Form
 - ▶ sollen den Leser beim Verstehen des Inhalts zu unterstützen.
- ▶ **Dokumente:** Texte mit Inhalt, Struktur und Form
 - ▶ z. B. Programme
- ▶ ein **Rat für Ihr weiteres Studium:**
 - ▶ viele Dokumente (PSE, Seminar, Bachelor-Arbeit, ...)
 - ▶ Finden geeigneter Struktur hilft auch eigenem Verständnis
 - ▶ deswegen: *früh damit beginnen, etwas aufzuschreiben*
(weil man dann über Struktur nachdenken muss)

- ▶ auch da spielt syntaktische Korrektheit eine Rolle,
- ▶ zumindest wenn Rechner im Spiel sind.
- ▶ Beispiele: **Auszeichnungssprachen** (engl. *markup language*)
 - ▶ Listen in $\text{\LaTeX}$
 - ▶ und ein klein bisschen Allgemeines zu $\text{\LaTeX}$
 - ▶ Tabellen in XHTML

Dokumente

L^AT_EX

XHTML

Eine Grenze unserer bisherigen Vorgehensweise

- ▶ basiert auf T_EX von Donald Knuth
<http://www-cs-faculty.stanford.edu/~knuth/>
- ▶ ausgesprochen „Tech“ (bzw. „Latech“)
- ▶ Textsatz-Programm
- ▶ in der Informatik sehr häufig verwendet wegen
 - ▶ hervorragendem automatischen Satz mathematischer Formeln
 - ▶ aus

$$\left[2 - \sum_{i=0}^k i 2^{-i} = (k+2) 2^{-k} \right]$$
 - ▶ wird

$$2 - \sum_{i=0}^k i 2^{-i} = (k+2) 2^{-k}$$

- ▶ Vorlesungsskript und Folien sind auch mit LaTeX gemacht
 - ▶ Diese Folie beginnt so:


```
\begin{frame}[fragile]
  \frametitle{\LaTeX}

  \begin{itemize}
    \item basiert auf \TeX{} von Donald Knuth
```

- ▶ Im Skript steht zum Beispiel
`\section{Struktur von Dokumenten}`
- ▶ woraus LaTeX die Zeile

7.2 STRUKTUR VON DOKUMENTEN

auf Seite 50 des Skriptes gemacht hat

- ▶ also
 - ▶ automatisch die passende Abschnittsnummer eingefügt
 - ▶ alles wurde in Großbuchstaben gesetzt
- ▶ Beachte: z. B. Schriftauswahl ist *nicht* in der Eingabe mit vermerkt ist.
- ▶ Diese Festlegung findet sich an anderer Stelle, und zwar an *einer* Stelle, an der das typografische Aussehen *aller* Abschnittsüberschriften (einheitlich) festgelegt ist.

```
\documentclass[11pt]{report}
% so schreibt man Kommentare
% dieser Teil heißt Präambel des Dokumentes
% Unterstützung für Deutsch,
% z.B. richtige automatische Trennung
  \usepackage[german]{babel}
  % Angabe des Zeichensatzes, in dem der Text ist
  \usepackage[latin1]{inputenc}
  % für das Einbinden von Grafiken
  \usepackage{graphicx}
\begin{document}
  % und hier kommt der eigentliche Text .....
\end{document}
```

- ▶ Eine Liste einfacher Punkte sieht in L^AT_EX so aus:

Eingabe	Ausgabe
<code>\begin{itemize}</code>	
<code>\item Inhalt</code>	• Inhalt
<code>\item Struktur</code>	• Struktur
<code>\item Form</code>	• Form
<code>\end{itemize}</code>	

- ▶ der dicke Punkt als Markierung ist *nicht* an der Stelle der Liste festgelegt.
- ▶ **Trennung** der Spezifikation von **Struktur** und der Spezifikation von **Form**
- ▶ Wenn man z. B. „–“ statt „•“ möchte, dann muss man an *einer* Stelle (in der Präambel) die Definition `\item` ändern.

- ▶ Gesucht: die formale Sprache L_{itemize} aller legalen Texte für Listen in LaTeX
- ▶ Gegeben: die formale Sprache L_{item} aller Texte, die hinter einem Aufzählungspunkt vorkommen dürfen.
(tun wir mal so ...)
- ▶ Dann
$$L_{\text{itemize}} = \{\backslash\text{begin}\{\text{itemize}\}\} \left(\{\backslash\text{item}\} L_{\text{item}} \right)^+ \{\backslash\text{end}\{\text{itemize}\}\}$$
- ▶ Problem: in Aufzählungspunkt wieder eine Liste erlaubt; also

$$\begin{aligned} L_{\text{item}} &= \dots L_{\text{itemize}} \dots \\ L_{\text{itemize}} &= \dots L_{\text{item}} \dots \end{aligned}$$

- ▶ Gleichungssystem lösbar? eindeutig lösbar?

Dokumente

L^AT_EX

XHTML

Eine Grenze unserer bisherigen Vorgehensweise

- ▶ HTML: Auszeichnungssprache, die man benutzt, wenn man eine WWW-Seite (be)schreibt.
- ▶ XHTML: sozusagen im wesentlichen eine noch striktere Variante von HTML
- ▶ Für beide formaler als für $\text{\LaTeX}$ festgelegt, wie syntaktisch korrekte solche Seiten aussehen.
- ▶ Das geschieht in einer sogenannten **document type definition**, kurz *DTD*.

```
<!ELEMENT table (caption?, thead?, tfoot?, (tbody+|tr+))>
<!ELEMENT caption %Inline;>
<!ELEMENT thead (tr)+>
<!ELEMENT tfoot (tr)+>
<!ELEMENT tbody (tr)+>
<!ELEMENT tr (th|td)+>
<!ELEMENT th %Flow;>
<!ELEMENT td %Flow;>
```

```
<!ELEMENT table (caption?, thead?, tfoot?, (tbody+|tr+))>
<!ELEMENT thead (tr)+>
<!ELEMENT tfoot (tr)+>
<!ELEMENT tbody (tr)+>
<!ELEMENT tr (th|td)+>
```

- ▶ **table, thead, tr**: Namen für formale Sprachen
- ▶ Bedeutung von **+** ε -freier Konkatenationsabschluss
- ▶ Bedeutung von **,** Produkt formaler Sprachen
- ▶ Bedeutung von **|** Vereinigung
- ▶ Fragezeichen ist neu, aber ganz einfach:

$$L^? = L^0 \cup L^1 = \{\varepsilon\} \cup L$$

- ▶ „optionales“ Auftreten eines Wortes aus L

- ▶ Mitteilung:

`<!ELEMENT tbody (tr)+ >` legt fest:

$$L_{\text{tbody}} = \{\text{<tbody>}\} \cdot L_{\text{tr}}^+ \cdot \{\text{</tbody>}\}$$

- ▶ Tabellenrumpf beginnt mit `<tbody>`, endet mit `</tbody>`, dazwischen eine beliebige positive Anzahl von Tabellenzeilen
- ▶ erste Zeile aus der DTD besagt:

$$L_{\text{table}} = \{\text{<table>}\} \cdot L_{\text{caption}}^? \cdot L_{\text{thead}}^? \cdot L_{\text{tfoot}}^? \cdot L_{\text{tbody}}^+ \cdot \{\text{</table>}\}$$

- ▶ Tabelle ist von Zeichenketten `<table>` und `</table>` umschlossen und enthält innerhalb in dieser Reihenfolge
 - ▶ optional eine Überschrift (*caption*),
 - ▶ optional einen Tabellenkopf (*table head*),
 - ▶ optional einen Tabellenfuß (*table foot*) und
 - ▶ eine beliebige positive Anzahl von Tabellenrumpfen.

```
<table>
  <tbody>
    <tr>  <td>1</td> <td>a</td>  </tr>
    <tr>  <td>2</td> <td>b</td>  </tr>
  </tbody>
</table>
```

Dokumente

L^AT_EX

XHTML

Eine Grenze unserer bisherigen Vorgehensweise

- ▶ eben mit Hilfe von Produkt und Konkatenationsabschluss formaler Sprachen präzise Aussagen gemacht.
- ▶ Das ging, weil
 - ▶ etwas von einer komplizierteren Art aus Bestandteilen einfacherer Art zusammengesetzt wurde.
- ▶ Aber manchmal: größere Dinge einer Art werden zusammengesetzt aus kleineren Bestandteilen *der gleichen Art*
- ▶ Beispiel: Listen, deren Aufzählungspunkte ihrerseits wieder Listen enthalten dürfen ...

- ▶ eben mit Hilfe von Produkt und Konkatenationsabschluss formaler Sprachen präzise Aussagen gemacht.
- ▶ Das ging, weil
 - ▶ etwas von einer komplizierteren Art aus Bestandteilen einfacherer Art zusammengesetzt wurde.
- ▶ Aber manchmal: größere Dinge einer Art werden zusammengesetzt aus kleineren Bestandteilen *der gleichen Art*
- ▶ Beispiel: Listen, deren Aufzählungspunkte ihrerseits wieder Listen enthalten dürfen ...

- ▶ anderes typisches Beispiel: korrekte Klammerungen (wie bei arithmetischen Ausdrücken)
- ▶ Bei einer syntaktisch korrekten Klammerung gibt es zu jeder Klammer auf „weiter hinten“ die „zugehörige“ Klammer zu.
- ▶ insbesondere legal:
 - ▶ mehrere korrekte Klammerungen hintereinander
 - ▶ zusätzliche Klammern um korrekte Klammerung außen herum
- ▶ schön wäre z. B. Beziehung von L_{Klammer} mit L_{Klammer}^* und mit $\{()\} \cdot L_{\text{Klammer}} \cdot \{()\}$
- ▶ aber wie? Gleichung?
 - ▶ lösbar?
 - ▶ wenn ja: eindeutig?
 - ▶ ein Thema des nächsten Kapitels ...

- ▶ Dokumente haben Inhalt, Struktur und Form.
- ▶ formale Sprachen helfen bei der Definition legaler Strukturen.