

# Computergrundlagen Textsatz mit L<sup>A</sup>T<sub>E</sub>X

Maria Fyta

Institut für Computerphysik  
Universität Stuttgart

Wintersemester 2012/13

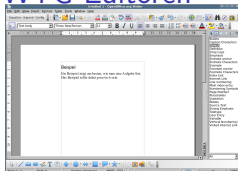
## Beispiel

Ein Beispiel zeigt am besten, wie man eine Aufgabe löst. Das Beispiel sollte dabei *generisch* sein.

- ▶ **Inhalt:** der Inhalt, also die Semantik des Textes  
*Hier: eine seltsame Definition von „Beispiel“*
- ▶ **Struktur:** Einteilung in logische Einheiten  
*Kapitel, Überschriften (hier „Beispiel“), Hervorhebungen („generisch“), Fußnoten, ...*
- ▶ **Layout:** wie soll der Text aussehen?  
*Blocksatz, Überschriften fett und etwas größer, Hervorhebungen kursiv, ...*

# Textverarbeitungssysteme

## WYSIWYG-Editoren



- ▶ z.B. MS Word, OO Writer
- ▶ „What you see is what you get“
- ▶ Layout direkt bei der Eingabe
- ▶ *Optik* im Vordergrund
- ▶ Struktur durch

## Markup-Sprachen

---

```
\documentclass{scrartcl}
```

```
\begin{document}  
\section{Beispiel}
```

Ein Beispiel zeigt am besten, wie man eine Aufgabe lösen sollte. Das Beispiel sollte dabei **generisch** sein.

```
\end{document}
```

---

- ▶ z.B.  $\text{\LaTeX}$ , (HTML)
- ▶ Dokument muss *kompiliert* werden
- ▶ Layout erst bei Kompilierung
- ▶ *Struktur* im Vordergrund
- ▶ Markup beschreibt Struktur

# Praktische Unterschiede

## MS Word/OO Writer

- ▶ Einfach bedienbar über GUI
- ▶ Nutzen von Formatvorlagen erfordert Disziplin
- ▶ Binäre Formate (.doc, ...)
- ▶ Integrierter Editor mit Rechtschreibprüfung, Druckvorschau, ...
- ▶ Nur eine mögliche Oberfläche
- ▶ Textsatz so gut wie Autor oder Vorlage

## L<sup>A</sup>T<sub>E</sub>X

- ▶ Erlernen der Satzsprache
- ▶ Strukturierung „natürlich“
- ▶ Textdateien
- ▶ Einzelprogramme Editor, Rechtschreibprüfung, Betrachter, ...
- ▶ Freie Wahl der Komponenten
- ▶ Textsatz mit Buchdruckqualität

# Textsatz

- ▶ Erstellen einer Druckvorlage für einen Text.
- ▶ Vom 15. – 19. Jahrhundert durch Setzen von Lettern in Kästen, daher die Bezeichnung „Textsatz“.



Dabei sind unter anderem zu beachten:

- ▶ Ausrichten der Absätze (z.B. Blocksatz)
- ▶ Verteilung der Absätze auf Seiten
- ▶ Worttrennung, um Lücken zu vermeiden
- ▶ Ligaturen („Trennung“ statt „Trennung“)
- ▶ Formelsatz
- ▶ Platzierung von Illustrationen

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse vel mauris. Cras quam tellus, interdum dapibus, mattis eu, pharetra vitae, neque. Nam semper mollis risus. Quisque sollicitudin pellentesque turpis. Aenean blandit, urna nec vehicula sollicitudin, ipsum justo condimentum dolor, vitae feugiat augue leo quis velit.	<b>desol.</b> Quisque sollicitudin ti ncidunt libero. Donec aliquet leo non tortor. Aenean vel massa quis nunc hendrerit vestibulum.
8	9

## TEX

- ▶ Name aus dem Griechischen, daher „Tech“ ausgesprochen
- ▶ Professionelles Textsatzsystem von Donald Knuth
- ▶ Entwickelt seit 1977, letzter Bugfix 2008

## L<sup>A</sup>TEX

- ▶ Strukturbeschreibungssprache von Leslie Lamport
- ▶ Basierend auf TEX als Satzsystem
- ▶ Variables Layout durch Dokumentklassen
- ▶ Baukastensystem: Kombinierbare Tools für Farben, Graphiken, Formeln usw.
- ▶ De-facto Standard im naturwissenschaftlichen Bereich
- ▶ TEX und L<sup>A</sup>TEX sind freie Software und Open Source
- ▶ „TEX“ wird häufig als Synonym für L<sup>A</sup>TEX gebraucht

## Verfügbarkeit

- ▶ In allen Linuxdistributionen als Pakete verfügbar
- ▶ Komplettdistribution T<sub>E</sub>Xlive: <http://www.tug.org/texlive>
- ▶ WYSIWYG-Editoren LyX: <http://www.lyx.org> und Kile (KDE)
- ▶ Speziell für Windows: MikT<sub>E</sub>X: <http://miktex.org> oder T<sub>E</sub>XnicCenter: <http://www.texniccenter.org>
- ▶ Weitere kommerzielle Editoren, z.B. Scientific Workplace

## Literatur

- ▶ L<sup>A</sup>T<sub>E</sub>X-Kurzanleitung:  
<http://www.tug.org/tutorials/lshort/l2kurz.pdf>
- ▶ „The Not So Short Introduction to L<sup>A</sup>T<sub>E</sub>X“: <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>
- ▶ H. Kopka: *L<sup>A</sup>T<sub>E</sub>X-Einführung Band 1*. Addison-Wesley, 2000.
- ▶ M. Goossens *et al*: *Der L<sup>A</sup>T<sub>E</sub>X-Begleiter*. Addison-Wesley, 2000.

## Was ist L<sup>A</sup>T<sub>E</sub>X ?

L<sup>A</sup>T<sub>E</sub>X ist ein sog. Satzsystem, mit dem Texte gesetzt werden können. Im Vergleich zu Textverarbeitungsprogrammen wie OpenOffice Writer werden Formatierungen wie Fettschrift oder Schriftart als expliziter Befehl angegeben. Um ein Dokument zu erzeugen, muss die erstellte Quelldatei zuerst durch L<sup>A</sup>T<sub>E</sub>X verarbeitet werden:

- ▶ Der Text wird mit `pdflatex beispiel.tex` kompiliert (früher: `latex beispiel.tex`)
- ▶ Erzeugt ein PDF (Portable Document Format) `beispiel.pdf` (früher: `beispiel.dvi`, L<sup>A</sup>T<sub>E</sub>Xs eigenes Format. Erst dann kann die erzeugte Datei (`mein_dokument.dvi` bzw. `mein_dokument.pdf`) betrachtet werden.)
- ▶ Daneben `beispiel.log` mit Informationen zur Verarbeitung und `beispiel.aux` mit internen Informationen
- ▶ Betrachten unter Linux mit `evince beispiel.pdf` [oder `xpdf`, `okular` (KDE), `acroread`]



# Text in L<sup>A</sup>T<sub>E</sub>X

- ▶ Kommentare werden durch „%“ eingeleitet.

---

```
% Dies ist ein Kommentar in LaTeX
```

---

- ▶ Leerzeichen, Zeilenumbrüche und Tabulatorzeichen werden normalisiert. Eine Leerzeile beginnt einen neuen Absatz.

---

Es ist egal, wie viele Leerzeichen, Tabulatoren oder Zeilenumbrüche man verwendet.

Die Zeilenlänge im Code ist unabhängig vom Layout. Eine Leerzeile beginnt einen neuen Absatz.

---

## Ergebnis

Es ist egal, wie viele Leerzeichen, Tabulatoren oder Zeilenumbrüche man verwendet.

Die Zeilenlänge im Code ist unabhängig vom Layout. Eine Leerzeile beginnt einen neuen Absatz.

Das Layout wird von L<sup>A</sup>T<sub>E</sub>X automatisch gemacht. Es erzeugt Blocksatz, trennt Wörter, erzeugt Ligaturen usw.

- ▶ Zum weiteren Textsatz werden Befehle verwendet.

---

Sonst verwendet man in `\LaTeX{}` Befehle.

---

## Ergebnis

Sonst verwendet man in L<sup>A</sup>T<sub>E</sub>X Befehle.

# Aufbau eines L<sup>A</sup>T<sub>E</sub>X-Dokuments

---

```
\documentclass{scrartcl}
\usepackage[ngerman]{babel}

\begin{document}
\section*{Beispiel} % Ein Beispielabschnitt
```

Präambel

Ein Beispiel zeigt am besten, wie man eine Aufgabe löst.  
Das Beispiel sollte dabei **\emph{generisch}** sein.

```
\end{document}
```

---

- ▶ **\documentclass** legt die *Layoutklasse* fest
- ▶ **Präambel** vor **\begin{document}**: Einbinden von Paketen, Setzen von Layout-Parametern, ...
- ▶ Inhalt bis **\end{document}**

## Aufbau einer L<sup>A</sup>T<sub>E</sub>X Quelldatei – 2

Das Grundlayout eines Dokuments wird durch seine Klasse festgelegt. Die gebräuchlichsten Klassen sind:

**article** — Artikel für wissenschaftl. Zeitschriften, kürzere Berichte usw.

**book** — Klasse für das Setzen von Büchern

**report** — Klasse für längere Berichte, die mehrere Kapitel beinhalten

## Aufbau einer L<sup>A</sup>T<sub>E</sub>X Quelldatei – 3

Zusätzlich zur Klasse können sog. Klassenoptionen angegeben werden:

**10pt|11pt|12pt** — Schriftgröße. L<sup>A</sup>T<sub>E</sub>X verwendet normalerweise eine Schrift der Größe 10pt.

**a4paper** — Seitengröße. Wird keine Seitengröße angegeben, wird die amerikanische letter-Größe verwendet.

**titlepage|notitlepage** — Titelseite und Zusammenfassung können getrennt oder gemeinsam auf einer Seite erscheinen.

**oneside|twoside** — Einseitiger oder zweiseitiger Druck.

**landscape** — Seitenorientierung wird auf quer gestellt.

## Aufbau einer L<sup>A</sup>T<sub>E</sub>X Quelldatei – 4

Pakete können zusätzliche Befehle bereitstellen, das Layout verändern usw. Sie werden üblicherweise direkt nach dem `\documentclass`- Befehl angefügt. Im deutschsprachigen Raum ist bspw. die direkte Eingabe von Umlauten über die Tastatur üblich. Dazu bindet man das Paket `inputenc` ein:

- ▶ `\usepackage[latin1]{inputenc}`

Für die deutsche Silbentrennung nach neuer Rechtschreibung nimmt man `ngerman` (für die alte Rechtschreibung `german`):

- ▶ `\usepackage{ngerman}`

Weitere nützliche Pakete sind `pdftex`, `graphicx` und `fontenc`. Die Verfügbarkeit von Paketen ist allerdings abhängig von der jeweiligen T<sub>E</sub>X-Installation.

## Aufbau einer L<sup>A</sup>T<sub>E</sub>X Quelldatei – 5

Die minimale Quelldatei sieht mit den bisherigen Erweiterungen nun so aus:

---

```
\documentclass[a4paper]{article}  
\usepackage[latin1]{inputenc}  
  \usepackage{ngerman}  
  \begin{document}  
  Hallo Welt! Wir k\u00f6nnen jetzt auch Umlaute benutzen.  
  \end{document}
```

---

## Absätze und Einrückungen

Normaler Text wird im Blocksatz gesetzt. Eine oder mehrere Leerzeilen führen zu einem neuen Absatz.

Absätze werden durch eine zusätzliche Einrückung gekennzeichnet.

Ist dies nicht erwünscht, kann vor dem Befehl

`\begin{document}` die Einrücktiefe auf 0mm gestellt werden:

```
\setlength{\parindent}{0mm}
```

...

```
\begin{document}
```

...

# Leerzeichen und Kommentare

## Leerzeichen

”Unsichtbare“ Zeichen wie das Leerzeichen, Tabulatoren und das Zeilenende werden von  $\text{\LaTeX}$  einheitlich als Leerzeichen behandelt. Mehrere aufeinanderfolgende Leerzeichen werden dabei als ein Leerzeichen verstanden.

## Kommentare

Das Prozentzeichen `%` dient als Kommentarbefehl. Alles, was sich bis zum Zeilenende hinter einem Prozentzeichen befindet, wird von  $\text{\LaTeX}$  als Kommentar interpretiert und erscheint nicht im fertigen Dokument:

```
% Dies ist eine Kommentarzeile
```



## Silbentrennung

L<sup>A</sup>T<sub>E</sub>X versucht immer, automatisch die passenden Silbentrennungen zu finden. Sollte dies einmal nicht gelingen, kann man an den entsprechenden Stellen manuell nachhelfen. Der Befehl \- innerhalb eines Wortes bewirkt, dass dieses Wort dieses eine Mal nur an den mit \- markierten Stellen getrennt werden darf:

Ein\-gabe\-datei

# Gliederung und Überschriften

- ▶ Gliederung des Dokuments in Kapitel, Abschnitte, usw.:

`\chapter{Kapitel}`

Kapitel, Abschnitte und Unterabschnitte sind nummeriert.

`\section{Abschnitt}`

`\subsection{Unterabschnitt}`

`\subsubsection{Unterunterabschnitt}`

Mit ein bisschen Text.

`\subsection*{Nicht nummerierter Unterabschnitt}`

Die Nummerierung kann unterdrueckt werden.

`\paragraph{Absatz mit Titel}`

Und ein bisschen Text im Absatz.

---

- ▶ Automatische Nummerierung, kann auch unterdrückt werden.
- ▶ Nicht jede Ebene existiert in allen Dokumentklassen.

## Ergebnis

### 1 Kapitel

Kapitel, Abschnitte und Unterabschnitte sind nummeriert.

#### 1.1 Abschnitt

##### 1.1.1 Unterabschnitt

Unterunterabschnitt

Mit ein bisschen Text.

##### Nicht nummerierter Unterabschnitt

Die Nummerierung kann unterdrueckt werden.

**Absatz mit Titel** Und ein bisschen Text im Absatz.

# Generisches Markup

- ▶ `\emph`: Hervorhebung, üblicherweise kursiv

---

```
So kann ich etwas  
\emph{hervorheben}!
```

---

Ergebnis

So kann ich etwas *hervorheben!*

- ▶ Stichpunkte:

---

```
Eine Auflistung:  
\begin{itemize}  
\item Ein Punkt  
\item Ein \emph{Punkt}  
  \begin{itemize}  
    \item ein Unterpunkt  
  \end{itemize}  
\end{itemize}
```

---

Ergebnis

Eine Auflistung:

- ▶ Ein Punkt
- ▶ Ein *Punkt*
  - ▶ ein Unterpunkt

- ▶ analog Aufzählung mit `enumerate`:

---

```
\begin{enumerate}  
\item Ein Punkt  
  \begin{enumerate}  
    \item ein Unterpunkt  
  \end{enumerate}  
\end{enumerate}
```

---

Ergebnis

1. Ein Punkt
  - 1.1 ein Unterpunkt

# Beschreibungslisten

Beschreibungslisten werden mit der description-Umgebung gesetzt:

---

```
\begin{description}
\item[Computer:] Große, unförmige bis
legante Kisten, die außer Krach und Elektromog nur bedingt
zen haben.
\item[Computernutzer:] Meist ebenso
schlau wie die Kiste, vor der ersitzt.
\end{description}
```

---

## Ergebnis

**Computer:** Große, unförmige bis legante Kisten, die außer Krach und Elektromog nur bedingt zen haben.

**Computernutzer:** Meist ebenso schlaue wie die Kiste, vor der ersitzt.

Alle beschriebenen Listen können auch verschachtelt werden, d. h. in einer itemize-Liste kann eine enumerate-Liste oder auch eine zweite itemize-Liste eingebettet werden. Die Listenzeichen werden automatisch angepasst.

# Tabellen

---

```
\begin{tabular}[t]{|r|cl|}  
  \hline  
  1   & 2   & 3   \\ \hline  
  eins & zwei & drei \\ \hline  
  rechts & & \\ \hline  
  zentriert & & \\ \hline  
  links \\ \hline  
\end{tabular}
```

---

## Ergebnis

1	2	3
eins	zwei	drei
rechts	zentriert	links

- ▶ Horizontale Ausrichtung wird durch String beschrieben:
  - ▶ r: rechtsbündig
  - ▶ l: linksbündig
  - ▶ C: zentriert
  - ▶ |: fügt eine vertikale Linie ein
- ▶ **\hline** fügt eine horizontale Linie ein
- ▶ **\\** beendet die (Tabellen-)Zeile

## Positionierung von Tabellen

$\LaTeX$  versucht immer, eine passende Aufteilung von Text und Tabelle zu finden. Dadurch kann es passieren, dass im fertigen Dokument der zu einer Tabelle gehörende Fließtext nicht immer direkt vor oder hinter derselben zu finden ist. Mit der `table`-Umgebung kann auf die Positionierung ein gewisser Einfluss ausgeübt werden. Dazu werden hinter dem Umgebungsbeginn in eckigen Klammern Positionsparameter definiert: `\table{}[t]`

Die Parameter haben folgende Bedeutung:

- `h` — here, "hier"
- `t` — top; Seitenbeginn
- `b` — bottom; Seitenende
- `p` — page; eigene Seite

Werden keine Parameter (und damit keine eckigen Klammern) angegeben, wird automatisch `[tbp]` angenommen.

$\LaTeX$  schiebt eine Tabelle solange nach hinten, bis sie in den Text passt. Der Befehl `\caption{...}` definiert eine Tabellenunterschrift, die zudem automatisch nummeriert wird. Mit `\listoftables` kann schließlich ein Tabellenverzeichnis ausgegeben werden.

## Zitate

Um längere Zitate kenntlich zu machen, fasst man sie in eine quote-Umgebung ein:

---

```
\begin{quote}
\emph{Freundlichkeit ist eine Sprache,
die Taube h\"oren und Blinde lesen k nnen.}
\hfill Mark Twain
\end{quote}
```

---

So sieht das Zitat dann aus:

*Freundlichkeit ist eine Sprache, die Taube hören und  
Blinde lesen können.* *Mark Twain*

# Visuelles Markup

---

```
\begin{center}\Large
  Ein \underline{schlechtes}
  Beispiel
\end{center}
\begin{flushright}\small
  Schlechtes \textbf{Layout}\!
  ist \textit{Arbeit} in \LaTeX.
\end{flushright}
```

---

Ergebnis

Ein schlechtes Beispiel

Schlechtes **Layout**  
ist *Arbeit* in L<sup>A</sup>T<sub>E</sub>X.

- ▶ flushright, flushleft, center: Ausrichtung
- ▶ \Huge, \large, \small, \tiny: Fontgrößen
- ▶ \textit, \textbf, \underline: kursiv, fett, unterstrichen

Visuelles Markup möglichst selten einsetzen!



# Befehle, Umgebungen und Deklarationen

- ▶ **Befehle:**

`\befehl[optionaler Param.]{notwendiger Param.}`

Ein Befehl kann beliebig viele Parameter haben

*Beispiel:* `\documentclass[12pt]{scrartcl}`

- ▶ **Umgebungen:**

`\begin{Umgebung}{Parameter}... \end{Umgebung}`

Umgebungen markieren Abschnitte und wirken nur auf das, was zwischen begin und end steht

*Beispiel:* `\begin{center} ... \end{center}`

- ▶ **Deklarationen:**

`{\befehl ...}`

Deklaration wirken nur innerhalb des aktuellen Abschnitts, in geschweiften Klammern oder zwischen begin und end

*Beispiel:* `{\small {\Large Gro"s} klein}`