

# Ausarbeitungen und Präsentationen mit LaTeX

Tobias Röhm (roehm@in.tum.de)

Proseminar Unix Tools  
TU München

24. Oktober 2005

# Agenda

- 1 LaTeX Grundlagen
- 2 LaTeX Start: Von der Idee zum Ausdruck
- 3 LaTeX Syntax
- 4 LaTeX Sprachelemente
- 5 Präsentationen mit LaTeX Beamer
- 6 Anhang

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

- System zum Setzen von Dokumenten
- L<sup>A</sup>T<sub>E</sub>X ist v. a. geeignet für große Dokumente (z. B. Bücher oder Diplomarbeiten), kann aber für beliebige Veröffentlichungen verwendet werden
- Ausgabeformate (Bsp): .ps, .pdf
- es gibt viele Pakete für L<sup>A</sup>T<sub>E</sub>X die den Funktionsumfang erweitern (z. B. für Mathematiker, Musiker, ...)
- weite Verbreitung in Wissenschaft und Industrie

# Geschichte von T<sub>E</sub>X

- Autor: Donald Knuth
- schreibt Buchreihe "The Art of Computer Programming" und ist unzufrieden mit der Druckqualität
- entwickelt das Textsatzsystem T<sub>E</sub>X (Beginn: 1977)
- T<sub>E</sub>X stellt sicher, dass ein Dokument immer gleich gesetzt wird - egal auf welchem Betriebssystem, Architektur oder Drucker
- T<sub>E</sub>X bietet nur Grundfunktionen, zur bequemen Benutzung werden Makros bzw. Makropackete benötigt

# Geschichte von T<sub>E</sub>X

- Autor: Donald Knuth
- schreibt Buchreihe “The Art of Computer Programming“ und ist unzufrieden mit der Druckqualität
- entwickelt das Textsatzsystem T<sub>E</sub>X (Beginn: 1977)
- T<sub>E</sub>X stellt sicher, dass ein Dokument immer gleich gesetzt wird - egal auf welchem Betriebssystem, Architektur oder Drucker
- T<sub>E</sub>X bietet nur Grundfunktionen, zur bequemen Benutzung werden Makros bzw. Makropackete benötigt

# Geschichte von T<sub>E</sub>X

- Autor: Donald Knuth
- schreibt Buchreihe “The Art of Computer Programming“ und ist unzufrieden mit der Druckqualität
- entwickelt das Textsatzsystem T<sub>E</sub>X (Beginn: 1977)
- T<sub>E</sub>X stellt sicher, dass ein Dokument immer gleich gesetzt wird - egal auf welchem Betriebssystem, Architektur oder Drucker
- T<sub>E</sub>X bietet nur Grundfunktionen, zur bequemen Benutzung werden Makros bzw. Makropackete benötigt

# Geschichte von T<sub>E</sub>X

- Autor: Donald Knuth
- schreibt Buchreihe “The Art of Computer Programming“ und ist unzufrieden mit der Druckqualität
- entwickelt das Textsatzsystem T<sub>E</sub>X (Beginn: 1977)
- T<sub>E</sub>X stellt sicher, dass ein Dokument immer gleich gesetzt wird - egal auf welchem Betriebssystem, Architektur oder Drucker
- T<sub>E</sub>X bietet nur Grundfunktionen, zur bequemen Benutzung werden Makros bzw. Makropackete benötigt

# Geschichte von T<sub>E</sub>X

- Autor: Donald Knuth
- schreibt Buchreihe “The Art of Computer Programming“ und ist unzufrieden mit der Druckqualität
- entwickelt das Textsatzsystem T<sub>E</sub>X (Beginn: 1977)
- T<sub>E</sub>X stellt sicher, dass ein Dokument immer gleich gesetzt wird - egal auf welchem Betriebssystem, Architektur oder Drucker
- T<sub>E</sub>X bietet nur Grundfunktionen, zur bequemen Benutzung werden Makros bzw. Makropackete benötigt



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

- L<sup>A</sup>T<sub>E</sub>X ist ein (das beliebteste) Makropacket für T<sub>E</sub>X
- Leslie Lamport hat L<sup>A</sup>T<sub>E</sub>X entwickelt,  
1985: Version 2.09
- aktuelle Version: L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> (second edition)

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

- L<sup>A</sup>T<sub>E</sub>X ist ein (das beliebteste) Makropacket für T<sub>E</sub>X
- Leslie Lamport hat L<sup>A</sup>T<sub>E</sub>X entwickelt,  
1985: Version 2.09
- aktuelle Version: L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> (second edition)

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

- L<sup>A</sup>T<sub>E</sub>X ist ein (das beliebteste) Makropacket für T<sub>E</sub>X
- Leslie Lamport hat L<sup>A</sup>T<sub>E</sub>X entwickelt,  
1985: Version 2.09
- aktuelle Version: L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> (second edition)

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

L<sup>A</sup>T<sub>E</sub>X unterscheidet im Gegensatz zu WYSIWYG- Programmen (Bsp. Office Textverarbeitung) zwischen logischem Design und Formatierung

Vorteile:

- Konzentration auf Inhalt
- Formatierung/ Setzen übernehmen Experten
- Format - Anpassungen nur an einer Stelle im Dokument
- konsistente Formatierung

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

L<sup>A</sup>T<sub>E</sub>X unterscheidet im Gegensatz zu WYSIWYG- Programmen (Bsp. Office Textverarbeitung) zwischen logischem Design und Formatierung

Vorteile:

- Konzentration auf Inhalt
- Formatierung/ Setzen übernehmen Experten
- Format - Anpassungen nur an einer Stelle im Dokument
- konsistente Formatierung

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

L<sup>A</sup>T<sub>E</sub>X unterscheidet im Gegensatz zu WYSIWYG- Programmen (Bsp. Office Textverarbeitung) zwischen logischem Design und Formatierung

Vorteile:

- Konzentration auf Inhalt
- Formatierung/ Setzen übernehmen Experten
- Format - Anpassungen nur an einer Stelle im Dokument
- konsistente Formatierung

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

L<sup>A</sup>T<sub>E</sub>X unterscheidet im Gegensatz zu WYSIWYG- Programmen (Bsp. Office Textverarbeitung) zwischen logischem Design und Formatierung

Vorteile:

- Konzentration auf Inhalt
- Formatierung/ Setzen übernehmen Experten
- Format - Anpassungen nur an einer Stelle im Dokument
- konsistente Formatierung

# Die Eingabedatei

- LaTeX verarbeitet sog. Eingabedateien (mit Texteditor erstellt, Endung .tex)
- Regeln für die Eingabedatei:
  - Leerzeichen (oder Zeilenende) beenden ein Wort (Anzahl ist egal)
  - Eine Leerzeile beendet einen Absatz (Anzahl ist egal)
  - LaTeX - Befehle beginnen mit einem “ \ “
  - Kommentare beginnen mit einem “ % “



## Ein einfaches Beispiel

```
\documentclass{article}
\usepackage{german}
\begin{document}
  Hier kommt der Text hin
\end{document}
```

# Dokumentenklasse festlegen

```
\documentclass{article}
\usepackage{german}
\begin{document}
  Hier kommt der Text hin
\end{document}
```

## Package einbinden

```
\documentclass{article}  
\usepackage{german}  
\begin{document}  
  Hier kommt der Text hin  
\end{document}
```

## Textkörper abgrenzen

```
\documentclass{article}  
\usepackage{german}  
\begin{document}  
  Hier kommt der Text hin  
\end{document}
```

## Verarbeitung der .tex- Datei

Die .tex- Datei wird mit dem Kommandozeilenbefehl “latex datei.tex“ bzw. “pdflatex datei.tex“ kompiliert (oder über entsprechende Funktionen einer Editorumgebung)

Dabei generiert LaTeX einige Dateien:

.pdf	pdf- Datei
.dvi	(device independent) Ausgabe in geräteunabhängiger Form
.log	Verarbeitungs- Logdatei (Fehler, Meldungen)
...	...

## LaTeX Fehlermeldungen

- Tritt während der Verarbeitung ein Fehler auf, so wird auf der Konsole eine Fehlermeldung ausgegeben (und auch in die log-Datei geschrieben).
- Meist handelt es sich um einfache Fehler wie Vertipper oder obligatorische Teile der Eingabedatei wurden vergessen (z. B. Dokumentenklasse)

## LaTeX Fehlermeldungen

- Tritt während der Verarbeitung ein Fehler auf, so wird auf der Konsole eine Fehlermeldung ausgegeben (und auch in die log-Datei geschrieben).
- Meist handelt es sich um einfache Fehler wie Vertipper oder obligatorische Teile der Eingabedatei wurden vergessen (z. B. Dokumentenklasse)

# Befehle

- Befehle veranlassen LaTeX einen bestimmten Text in einer (durch den Befehl festgelegten) Art zu setzen (textgenerierende Befehle) oder ändern die aktuellen Einstellungen
- Syntax: `\befehlsname [optionen] {parameter}`
- Beispiele:

`\today` gibt das aktuelle Datum aus

`\documentclass` lädt entsprechende Dokumentklasse



# Umgebungen

- Umgebungen beeinflussen die Art, wie ein Text gesetzt wird (Formatierung, ...)
- Diese Einstellungen gelten für einen bestimmten Bereich.
- Syntax:  
`\begin{umgebungsname}`  
`\end{umgebungsname}`
- Beispiele:  
equation            Formel  
verbatim            gibt Text aus ohne zu interpretieren

# Titel (-seite)

- Ein Dokumententitel kann mit LaTeX automatisch erzeugt werden.

- Vorher müssen die Daten dafür eingegeben werden:

<code>\title{Titel}</code>	Dokumententitel
<code>\author{Autor}</code>	Autor(en) (mehrere mit <code>\and</code> trennen)
<code>\date{Datum}</code>	Datum

- Der Titel wird mit dem Befehl `\maketitle` erstellt

## Gliederung/ Inhaltsverzeichnis

- In LaTeX gibt es folgende Gliederungsbefehle:  
part, chapter, section, subsection, subsection
- Syntax: `\chapter{Kapitel 1}`
- Ein Gliederungsbefehl hat folgende Auswirkungen:
  - Setzen der Überschrift entsprechend Dokumentenklasse
  - Gliederungsinformation wird in generierten Teilen (z. B. Inhaltsverzeichnis) verwendet

# Gliederung/ Inhaltsverzeichnis

- Besonderheiten:
  - In der Dokumentenklasse article gibt es kein “chapter“
  - “part“ dient der Gliederung von sehr langen Dokumenten und hat auf die Nummerierung keine Auswirkung
  - Damit das Inhaltsverzeichnis aktualisiert wird muss LaTeX zweimal laufen (.toc Datei)
- Inhaltsverzeichnis erstellen: `\tableofcontents`

# Formatierung

- Aufgrund der LaTeX Philosophie (log. Design) sollte Formatierung nur in selbstdefinierten Befehlen vorkommen (Bsp: Befehl oder Umgebung für Computerprogramme definieren und dort zentral Formatierung festlegen)
- Mit dem `\emph{text}`- Befehl kann LaTeX mitgeteilt werden, dass "text" hervorgehoben werden soll (die Formatierung übernimmt die Dokumentenklasse)

# Formatierung

- LaTeX Befehle zur Formatierung:

Format	Befehl
<i>kursiv</i>	<code>\textit{...}</code>
<b>fett</b>	<code>\textbf{...}</code>
Roman- Schrift	<code>\textrm{...}</code>
TypeWriter	<code>\texttt{...}</code>
Farbe	<code>textcolor{red}{text}</code>

## Formelumgebungen

- In LaTeX gibt es mehrere Formelumgebungen:

Umgebung	Kurzform	Erklärung
math	$\backslash(\dots\backslash)$ $\$ \dots \$$	normale Formel (im Text)
displaymath	$\backslash[ \dots \backslash]$	abgesetzte Formel ohne Nummer
equation		abgesetzte Formel mit Nummer

- Eine Buchstabenfolge in einer Formelumgebung wird von LaTeX als Produkt interpretiert (Bsp:  $abc$ )  
Um in eine Formel Text einzufügen kann man den mbox-Befehl ( $\backslashmbox{\text{text}}$ ) benutzen (Bsp:  $abc$ )

# Symbole und griech. Buchstaben

In LaTeX kann gibt es für viele Symbole einen Befehl, mit dem man sie ganz einfach einsetzen kann (diese Befehle sind nur in Formelumgebungen zulässig)

Beispiele:

<code>\alpha</code>	$\alpha$	griechische Kleinbuchstaben: <code>\name</code>
<code>\Omega</code>	$\Omega$	griechische Großbuchstaben: <code>\Name</code>
<code>\subset</code>	$\subset$	Teilmenge- Zeichen
<code>\Rightarrow</code>	$\Rightarrow$	“Daraus folgt“- Zeichen
<code>\infty</code>	$\infty$	unendlich
<code>\sum</code>	$\sum$	Summe
...	...	...



## einfache Strukturen

- Brüche:  $\frac{1}{1+x}$  (`\frac{Zaehler}{Nenner}`)
- Wurzeln:  $\sqrt[n]{2}$  (`\sqrt[n]{ausdruck}`)
- Hochstellung:  $x^2$  (`x^{Exponent}`)
- Tiefstellung:  $x_2$  (`x_{Index}`)

Diese Elemente können auch verschachtelt werden:

$$x_t^{\sqrt{2}} \quad (\text{x}_{\text{t}}^{\sqrt{2}})$$

## unsichtbare Zeichen

- Zeilenumbruch erzwingen: “\\”
- Leerzeichen einfügen: “\ ”
- nicht trennbarer Wortzwischenraum: “wort~wort”

## Bilder einbinden

Um Bilder einzubinden kann das graphics- Packet (`\usepackage{graphics}`) verwendet werden.

Syntax: `\includegraphics{bild.jpg}`  
(Bild liegt im gleichen Verzeichnis)



# einfache Aufzählung

```
\begin{itemize}  
\item Punkt 1  
\item Punkt 2  
\end{itemize}
```

# nummerierte Aufzählung

```
\begin{enumerate}  
\item Punkt 1  
\item Punkt 2  
\end{enumerate}
```

## Sonstiges

Fußnoten<sup>1</sup>: `\footnote{Fu"snote}`

---

<sup>1</sup>Fußnote

# Deutsch in LaTeX

- Da LaTeX ursprünglich im englischen Sprachraum entstanden ist, muss es für die deutsche Sprache angepasst werden.
- In der Regel reicht es, einfach das "german"- Packet einzubinden, das diese Anpassungen im Hintergrund vornimmt.
- Deutsche Sonderzeichen müssen dann mit " markiert werden (Bsp: "a für ä, "s für ß, "' für ")

# LaTeX Beamer

- LaTeX Beamer ist eine Klasse, mit der einfach Präsentationen mit LaTeX erstellt werden können
- Einbinden von LaTeX Beamer:  
Dokumentenklasse    beamer    `\documentclass{beamer}`  
Template auswählen    Stadtname    `\usetheme{Warsaw}`
- Text, der auf Folien erscheinen soll, muss in `frame-` Umgebung stehen.
- Sonst kann LaTeX Beamer (fast) wie LaTeX benutzt werden



## Ausblick - mehr LaTeX

- Befehle und Umgebungen selbst definieren
- BibTeX - Tool zur Verwaltung von Quellen
- MakeIndex - Tool zur automatischen Indexgenerierung
- weitere nützliche Pakete
  - Paket listings (für Sourcecode)  
<http://www.ctan.org/info?id=listings>
  - Pakete von AMS (American Mathematical Society - Pakete für Mathematiker)  
<http://www.ams.org/tex/amslatex.html>
  - ...
- ...

## Links und Quellen

- Leslie Lamport: Das LaTeX Handbuch (Addison- Wesley)
- Wikipedia  
<http://www.wikipedia.de>
- LaTeX Project  
<http://www.latex-project.org/>
- LaTeX Beamer Homepage  
<http://latex-beamer.sourceforge.net/>
- Packet listings (für Sourcecode)  
<http://www.ctan.org/info?id=listings>
- AMS American Mathematical Society (Mathe- Pakete)  
<http://www.ams.org/tex/amslatex.html>

# Ausarbeitungen und Präsentationen mit LaTeX

Tobias Röhm (roehm@in.tum.de)

Proseminar Unix Tools  
TU München

24. Oktober 2005