

# Einführung in wissenschaftliches Schreiben: Einführung in $\text{\LaTeX}$

Éva Mújdricza-Maydt  
mujdricza@cl.uni-heidelberg.de

Institut für Computerlinguistik, Universität Heidelberg

Sommersemester 2019

Vielen Dank an alle Doktoranden, die mir freundlicherweise Materialien für diese Folien zur Verfügung gestellt haben.

- ① Einführung
- ② Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- ③ Spezielle Pakete
- ④ Umgebungen
- ⑤ Formatierungen
- ⑥ Strukturierung
- ⑦ Mathematikmodus
- ⑧ Zusammenfassung

- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen
- 5 Formatierungen
- 6 Strukturierung
- 7 Mathematikmodus
- 8 Zusammenfassung

$\text{T}_{\text{E}}\text{X}$  ist ein Textsatzsystem zur Erzeugung von typographisch hochwertigen Dokumenten, hauptsächlich im naturwissenschaftlichen Bereich

$\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$  ist eine Sammlung von arbeitserleichternden Makros für  $\text{T}_{\text{E}}\text{X}$

# Prinzipien

Unterschiede zu Textverarbeitungsprogrammen wie beispielsweise Microsoft Word:

- $\LaTeX$  ist eine Auszeichnungssprache wie HTML (*markup language*), keine Programmoberfläche  $\rightarrow$  kein WYSIWYG
- $\TeX$ - bzw.  $\LaTeX$ -Dateien sind *plain text*, das heißt sie bestehen lediglich aus Buchstaben, Zahlen und Sonderzeichen und können mit jedem Texteditor geschrieben werden, wobei es einige gibt, die speziell auf  $\LaTeX$  ausgerichtet sind (z.B. TeXstudio, TeXShop)
- Aus  $.tex$ -Dateien können jedoch eine Vielzahl von Formaten generiert werden: PS, PDF, HTML, DVI, TXT, ...

# Motivation: Warum $\text{\LaTeX}$ (nicht) verwenden?

Einige der Nachteile von  $\text{\LaTeX}$

- geringere Benutzerfreundlichkeit, steilere Lernkurve
- Layouts lassen sich nur geringfügig variieren, größere Abweichungen sind nur mit viel Aufwand möglich (Definition einer eigenen Dokumentklasse)

# Motivation: Warum $\text{\LaTeX}$ (nicht) verwenden?

## Einige der Vorteile von $\text{\LaTeX}$

- Produziert standardmäßig typographisch hochwertige Dokumente, unter anderem Artikel, Briefe, Bücher. . .
- Querverweise, Nummerierungen, Inhaltsverzeichnis, Bibliographie etc. können automatisch erzeugt und aktualisiert werden
- $\text{\LaTeX}$ -Dokumente sind plattformunabhängig
- Auch lange, aufwändige Dokumente lassen sich problemlos und mit vergleichsweise geringem Ressourcenverbrauch bewältigen
- $\text{\LaTeX}$  ist kostenlos, hat exzellente Dokumentation
- . . .

# Grundgerüst einer L<sup>A</sup>T<sub>E</sub>X-Datei

1. `\documentclass{article}`
2. `\usepackage[ngerman]{babel}`
3. `\title{Die Henne und das Ei}`
4. `\author{Charles Dickens}`
5. `\date{23. August 2010}`
  
6. `\begin{document}`
7.     `\maketitle`
8.     `[...]`
9. `\end{document}`

- 1 – 5 Die sogenannte **Präambel**, Definition der **Dokumentklasse**, Angabe von Metadaten und Einbindung von **Paketen**
- 7 – 8 Hier wird der eigentliche Inhalt des Dokuments geschrieben
- 6, 9 Definiert eine **Umgebung**, hier des Dokuments `document`



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

## Textzeichen

a...z A...Z 0...9

. : ; , ? ! ' ' ( ) [ ] - / \* @ + =

## Steuerzeichen

\$ & % # \_ { } ~ ^ " \ | < >

## Beispiel

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

- L<sup>A</sup>T<sub>E</sub>X-Befehle beginnen mit einem *backslash* \
- worauf der Name des Befehls folgt, hier `usepackage`
- optionale Parameter werden in eckigen Klammern [] übergeben, hier `ngerman`
- notwendige Parameter in geschweiften Klammern {}, hier `babel`

# Dokumentklasse

```
\documentclass{...}
```

- die Dokumentklasse kennzeichnet das Format für das gesamte Dokument<sup>1</sup>
- Typische Klassen: book, report, article, letter, beamer, ...
- Typische Optionen: a4paper, 10pt, 11pt, 12pt, oneside, twoside, landscape ...
- Beispiel:

```
\documentclass[a4paper,11pt,landscape]{article}
```

---

<sup>1</sup>diese Folien wurden mit der Dokumentklasse beamer erstellt

# Dokumentklassen

- Die Dokumentklassen `book`, `article` und `report` sind im Hinblick auf US-amerikanischen Textsatzkonventionen und Papierformaten entstanden
- Ein flexibleres, alternatives Layout bieten die KOMA-Script-Klassen `scrartcl`, `scrreprt`, `scrbook`

# Präambel

```
\documentclass{...}  
\usepackage[ngerman]{babel}  
\title{Die Henne und das Ei}  
\author{Charles Chickens}  
\date{10. April 2019}  
\begin{document}  
\maketitle
```

- die Präambel ist der Bereich zwischen der Angabe der Dokumentklasse und der Dokumentumgebung
- es können Erweiterungspakete geladen werden (`\usepackage`)
- man kann Titel, Autor und Datum des Dokuments festlegen (mit `\maketitle` lässt sich daraus eine Titelseite erzeugen)
- ...

demo - ein erstes L<sup>A</sup>T<sub>E</sub>X-Dokument

- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen
- 5 Formatierungen
- 6 Strukturierung
- 7 Mathematikmodus
- 8 Zusammenfassung

## Von .tex zu .pdf

Um aus einer  $\text{\LaTeX}$ -Datei ein druckbares Dokument zu erzeugen, gibt es im Wesentlichen zwei Möglichkeiten:

Kommandozeile:

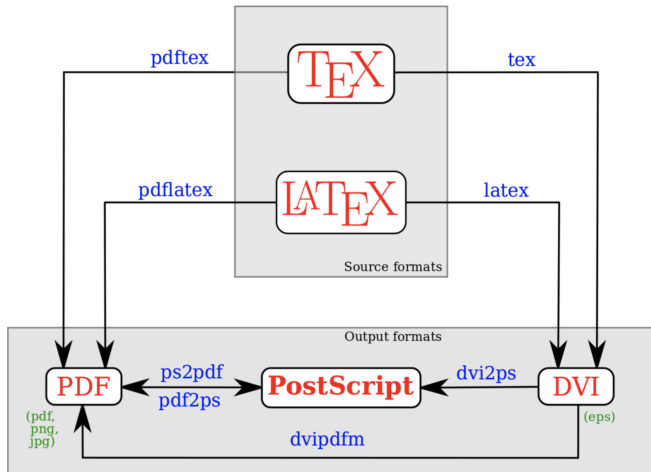
- `pdflatex meinedatei.tex`  
→ `meinedatei.pdf`

spezieller Editor oder Plugin:

- TeXstudio
- Texmaker
- Eclipse using TeXlipse
- ...



# Von .tex zu .pdf



Wo bekommen wir diese Programme her? Das ist je nach Betriebssystem sehr unterschiedlich:

- Windows: MiKTeX, <http://miktex.org/>
- Mac: MacTeX, <http://www.tug.org/mactex/>
- Linux: TeX Live<sup>2</sup>, <http://tug.org/texlive/>

Ist eine dieser Distributionen installiert, stehen auf der Kommandozeile die vorher genannten Befehle zur Verfügung bzw. fähige Editoren können benutzt werden.

---

<sup>2</sup>Am besten über eine Paketverwaltung installieren

# Fehlermeldungen

Beim Übersetzen von `.tex`-Dateien gibt  $\text{\LaTeX}$  oft Fehlermeldungen aus. Folgende Fehlermeldungen können u.a. auftreten:

`warnings` z.B. Fehlende Fonts, oft kein gravierendes Problem

`badboxes` z.B. `Overfull \hbox`, zeigt Probleme bei Zeilenumbrüchen an

`errors` Beginnen mit `!`, treten auf, wenn  $\text{\LaTeX}$  etwas nicht interpretieren oder ausführen kann

$\text{\LaTeX}$  generiert nicht nur das Ausgabe-PDF, sondern auch eine `.log`-Datei, in der steht, welche Schritte ausgeführt wurden und wo Fehlermeldungen aufgetreten sind.

# Fehlermeldungen

## Fehlermeldungen verstehen

- Errors führen dazu, dass  $\text{\LaTeX}$  die Fehlerbenachrichtigung anzeigt und den Übersetzungsprozess anhält. Es wird keine Ausgabedatei generiert.
- Oft enthält die Fehlermeldung eine Zeilenangabe, die helfen kann, den Fehler zu finden
- Häufig auftretende Fehler sind:
  - Tippfehler
  - Fehlende `}` oder unvollständige Umgebung
  - Fehlender Mathemodus
  - Paket nicht geladen oder nicht installiert

# Umgang mit Fehlermeldungen

## Fehlermeldungen vermeiden/Troubleshooting

- Highlight-Modus des Editors verwenden, dieser zeigt z.B. fehlende Klammern
- Viele Editoren bieten automatische Vervollständigung an
- Große Dateien vermeiden, lieber ein Master-Dokument verwenden und einzelne Kapitel mit `\input` einfügen
- `pdflatex` während des Schreibprozesses oft laufen lassen, das vermeidet langwierige Fehlersuche am Ende

# Hilfe zur Selbsthilfe I

Hilfen/Literatur zu Latex allgemein:

- Website der Tex User Group (Downloads, umfassende Linksammlung zu Dokumentationen und weiteren Usergroups, PracTeX Journal) <http://www.tug.org>
- Homepage (Links zu Dokumentation, FAQ) <http://www.latex-project.org>
- Comprehensive T<sub>E</sub>X Archive Network (CTAN) (Download von Paketen inkl. Dokumentation) <http://www.ctan.org>
- Lshort (L<sup>A</sup>T<sub>E</sub>X-Einführung von Tobias Oetiker, in mehreren Sprachen erhältlich) <http://www.ctan.org/tex-archive/info/lshort/>
- The L<sup>A</sup>T<sub>E</sub>X wiki book (nutzergeneriertes Handbuch mit vielen hilfreichen Code-Beispielen) <http://en.wikibooks.org/wiki/LaTeX>

## Hilfe zur Selbsthilfe II

- Befehlsreferenz  
<http://www.weinelt.de/latex/>
- Symbolreferenz  
<http://detexify.kirelabs.org>
- Latex Community (User Forum)  
<http://www.latex-community.org>

Dokumentation zu einzelnen Paketen:

- Bib $\text{T}_\text{E}\text{X}$  – kurze Einführung von Jürgen Fenn  
*Managing Citations and your Bibliography with Bib $\text{T}_\text{E}\text{X}$ (2006)*  
<http://www.tug.org/pracjourn/2006-4/fenn/fenn.pdf>
- natbib Handbuch  
über CTAN.org erhältlich.  
Z.B. <http://sunsite.informatik.rwth-aachen.de/ftp/pub/mirror/ctan/macros/latex/contrib/natbib/natbib.pdf>

## Hilfe zur Selbsthilfe III

- natbib Referenz  
<http://merkel.zoneo.net/Latex/natbib.php>
- gnuplot
  - Dokumentation:  
[http://www.gnuplot.info/docs\\_4.4/gnuplot.pdf](http://www.gnuplot.info/docs_4.4/gnuplot.pdf)
  - gnuplot und  $\LaTeX$ :  
<http://www.gnuplot.info/files/tutorial.pdf>



- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete**
- 4 Umgebungen
- 5 Formatierungen
- 6 Strukturierung
- 7 Mathematikmodus
- 8 Zusammenfassung

# Wichtige Pakete

`\usepackage [option] {paketname}`

`babel` Sprachspezifische Einstellungen (Optionen: `german`, `ngerman`, ...)

`inputenc` Zeichenkodierung (Optionen: `utf8`, `latin1`, ...)

`url` URLs und Links

`graphicx` Einbinden von Bilddateien (`jpg`, `png`, `pdf`, ...)

`tikz` Grafiken (z.B. Bäume, Diagramme, etc.)

`listings` Programmcode

`AVM` Attribut-Wert-Matrizen

`Covington` (linguistische) Beispiele

`tree-dvips` Bäume, Linien, etc.

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

L<sup>A</sup>T<sub>E</sub>X ist ursprünglich auf amerikanisches Englisch ausgerichtet. Folgende Probleme ergeben sich, wenn man Text in anderen Sprachen schreiben will:

- 1 Umgang mit unterschiedlichen orthographischen Regeln, z.B. Silbentrennung am Zeilenende
- 2 Umgang mit anderen typographischen Konventionen (z.B. Anführungszeichen)
- 3 Kapitelüberschriften etc. in der Sprache des Dokuments
- 4 Sprachliche Zeichen, die nicht im Standardzeichensatz von L<sup>A</sup>T<sub>E</sub>X enthalten sind (z.B. Umlaute)

Für diese Probleme wurde das Paket `babel` entwickelt

## Das Paket babel

- babel bietet Sprachunterstützung für viele, hauptsächlich europäische Sprachen (siehe Liste in Dokumentation)
- Das Paket lädt automatisch die Trennungsregeln für die gewünschte Sprache und erzeugt entsprechende Kapitelüberschriften
- Einbinden des Pakets: z.B. `\usepackage[ngerman]{babel}` für neue deutsche Rechtschreibung (`german` für alte Rechtschreibung)
- In der Präambel können auch mehrere Sprachen angegeben werden. Umschalten kann man mit `\selectlanguage{language}`

# Umlaute und Anführungszeichen

(Ohne spezielle Spracheinstellungen)

**Anführungszeichen** Deutsche Anführungszeichen („ “) können mit `\glqq \grqq` oder mit `"‘ ”’` erzeugt werden.

**Umlaute** Sehr umständliche Eingabe ohne `babel` :

`\"a` → ä

Umlaute mit `babel`:

`"a` → ä

**scharfes s** Unter `babel` mit `\ss`

# Kodierungen I

Die Kodierung betrifft sowohl die Eingabe- als auch die Ausgabedatei.

## 1. Eingabekodierung

- `\usepackage[<encoding>]{inputenc}`: gibt die Kodierung der `.tex`-Datei an.
- Wird, wie hier, `utf8` als Kodierung verwendet, können auch im Quelltext Sonderzeichen (z.B. Umlaute) verwendet werden
- Das ist komfortabler, funktioniert aber nur, wenn die Kodierung vom System unterstützt wird

## 2. Ausgabekodierung

- `\usepackage[T1]{fontenc}` gibt die Kodierung des Ausgabefonts an
- T1 lädt beispielsweise Schriften, die über die verwendeten Umlaute/Akzente für die meisten europäischen Sprachen verfügen.
- Für andere Schriftsysteme müssen andere Kodierungen geladen werden

## Weitere Pakete finden

- Die meisten Pakete gibt es über das CTAN<sup>3</sup>  
<http://www.ctan.org>
- latex4ling listet eine Reihe von Paketen, die man als Linguist braucht  
<http://www.essex.ac.uk/linguistics/external/clmt/latex4ling/>
- ams-latex bietet viele Erweiterungen zum Setzen mathematischer Texte  
<http://www.ams.org/tex/amslatex.html>

---

<sup>3</sup>the Comprehensive T<sub>E</sub>X Archive Network



- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen**
- 5 Formatierungen
- 6 Strukturierung
- 7 Mathematikmodus
- 8 Zusammenfassung

# Dokumentumgebung

```
\begin{document}
```

Der eigentliche Inhalt eines Dokuments steht in der Dokumentumgebung. Beliebig viele Leerzeichen werden als `\emph{ein}` Leerzeichen behandelt.

Einen neuen Absatz erhält man mittels einer Leerzeile. Kommentare beginnen mit einem % blöden % oder besser: einem lehrreichen Beispiel Prozentzeichen (`\%`).

```
\end{document}
```

Der eigentliche Inhalt eines Dokuments steht in der Dokumentumgebung. Beliebig viele Leerzeichen werden als *ein* Leerzeichen behandelt.

Einen neuen Absatz erhält man mittels einer Leerzeile. Kommentare beginnen mit einem Prozentzeichen (%).

# LaTeX-Umgebungen

## Beispiel

```
\begin{itemize}
\item LaTeX-Umgebungen definieren Bereiche, in denen
    besondere Regeln für die Formatierung gelten
\item ...
\end{itemize}
```

- LaTeX-Umgebungen definieren Bereiche, in denen besondere Regeln für die Formatierung gelten
- eine Umgebung beginnt mit `\begin{umgebungsname}` und endet mit `\end{umgebungsname}` (hier: `itemize`)
- in manchen Umgebungen stehen zusätzliche Befehle zur Verfügung (hier: `\item`)

# Wichtige Umgebungen – Listen

① nummerierte Listen

② Aufzählungslisten

③ Beschreibungslisten

- nummerierte Listen

- Aufzählungslisten

- Beschreibungslisten

`enumerate` nummerierte Listen

`itemize` Aufzählungslisten

`description` Beschreibungslisten

```
\begin{enumerate}
\item nummerierte Listen
\item Aufzählungslisten
\item Beschreibungslisten
\end{enumerate}
```

```
\begin{itemize}
\item nummerierte Listen
\item Aufzählungslisten
\item Beschreibungslisten
\end{itemize}
```

```
\begin{description}
\item[enumerate] nummerierte Listen
\item[itemize] Aufzählungslisten
\item[description] Beschreibungslisten
\end{description}
```

## Wichtige Umgebungen – Tabellen

```
\begin{table}
  \begin{tabular}{|r|l|}
    \hline
    7C0 & hexadezimal \\
    3700 & oktal \\
    11111000000 & bin\"ar \\
    \hline\hline
    1984 & dezimal \\
    \hline
  \end{tabular}
\caption{Neunzehnhundertvierundachtzig}
\end{table}
```

7C0	hexadezimal
3700	oktal
11111000000	binär
1984	dezimal

**Tabelle:** Neunzehnhundertvierundachtzig

# Wichtige Umgebungen – Abbildungen

```
\usepackage{graphicx} % in der Präambel
```

```
\begin{figure}
```

```
\includegraphics[scale=0.15]{fig/Buchdrucker-1568.png}
```

```
\caption{Buchdruck im 16. Jahrhundert}
```

```
\end{figure}
```

**Abbildung:** Schematischer L<sup>A</sup>T<sub>E</sub>X-Code für untere Abbildung



**Abbildung:** Buchdruck im 16. Jahrhundert

- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen
- 5 Formatierungen**
- 6 Strukturierung
- 7 Mathematikmodus
- 8 Zusammenfassung

# Textformatierung

Formatierungen wie Hervorheben oder Unterstreichen werden in  $\text{\LaTeX}$  durch Befehle vorgenommen

- `\textbf{fetter Text}` **fetter Text**
- `\textit{kursiver Text}` *kursiver Text* (italics)
- `\texttt{feste Laufweite}` feste Laufweite (typewriter)
- `\underline{unterstrichen}` unterstrichen



`\Huge`

`\huge`

`\LARGE`

`\Large`

`\large`

`\normalsize`

`\small`

`\footnotesize`

`\scriptsize`

`\tiny`

In  $\text{\LaTeX}$  gibt es verschiedene Arten von Befehlen:

- **Befehl + Option**,  
z. B. `\textbf{fetter Text}` oder
- **Schalter**, z. B. `\Large`. Schalter haben so lange Gültigkeit bis sie wieder aufgehoben werden, hier durch `\normalsize`.

# Hervorhebungen und Fußnoten

- *Emphatischer Text*<sup>4</sup> lässt sich mittels `\emph{Emphatischer Text}` erzeugen
- Eine Fußnote erzeugt man mittels `\footnote{d.h. hervorgehobener Text}`

---

<sup>4</sup>d.h. hervorgehobener Text

# Sonderzeichen

- Steuerzeichen von  $\text{\LaTeX}$  müssen umgangen werden, wenn sie als Zeichen im Text vorkommen sollen
  - Die meisten Zeichen werden mit `\` vor dem Zeichen geschrieben, z.B. `\&`, `\$`, `\%`  $\rightarrow$  `&`, `$`, `%`
  - Manchmal hat auch `\` Zeichen eine Sonderbedeutung, z.B. `\\`, hier brauchen wir den Befehl `\textbackslash`
- Für Sonderzeichen, die nicht im Inventar von  $\text{\LaTeX}$  sind, werden auch Befehle verwendet, z.B. `\texteuro` für  $\text{\text{€}}$ <sup>5</sup>
- eine Übersicht gibt es hier: [http://en.wikibooks.org/wiki/LaTeX/Special\\_Characters](http://en.wikibooks.org/wiki/LaTeX/Special_Characters)

---

<sup>5</sup>erfordert `\usepackage{textcomp}`

## Abstände und Umbrüche

- Der Abstand nach „.“ wird automatisch vergrößert. Das kann mit `.~` oder `.\` abgestellt werden. `~` verhindert zusätzlich einen Zeilenumbruch
- Einen manuellen Zeilenumbruch erzeugt man mit `\\`
- Größerer Abstand zwischen Zeilen: `\bigskip`
- Die Befehle `\hspace{<Länge>}` und `\vspace{<Höhe>}` erzeugen horizontalen und vertikalen Abstand.  
Beispielangaben: `0.5cm` oder `1in`

Es empfiehlt sich, sparsam mit eigenen Abstandsdefinitionen umzugehen, da  $\text{\LaTeX}$  (meistens) von selbst gut leserlichen und stimmig angeordneten Output erzeugt.

# Trennung

- Manchmal hat  $\LaTeX$  Probleme mit der Trennung von Wörtern, was zu badboxes führen kann
- Trennungsvorgaben lassen sich auch manuell machen:
- `\-` in einem Wort markiert mögliche Trennungspunkte  
`Do\ - nau\ - dampf\ - schiff\ - fahrts...`
- In der Präambel können Trennungsvorgaben mit `textbackslash hyphenation{}` angegeben werden  
`\hyphenation{Do-nau-dampf-schiff-fahrts...}`

- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen
- 5 Formatierungen
- 6 Strukturierung**
- 7 Mathematikmodus
- 8 Zusammenfassung

# Kapitel und Überschriften

- Welche Typen von Überschriften es gibt, hängt von der Dokumentklasse ab
- bei `article` sind das:
  - `\section{...}`
  - `\subsection{...}`
  - `\subsubsection{...}`
- Der Befehl `\tableofcontents` erzeugt automatisch ein Inhaltsverzeichnis

# Abbildungen & Tabellen

- Abbildungen und Tabellen können wie auch `equation` automatisch mit einer Nummerierung versehen werden, hierfür dienen die Umgebungen `figure` bzw. `table`
- Diesen Umgebungen kann man mittels `\caption{...}` eine Bild- oder eine Tabellenbeschreibung hinzufügen (sie erscheint unterhalb der Abbildung/Tabelle)
- Abbildungs- und Tabellenverzeichnis können mit `\listoffigures` bzw. `\listoftables` erzeugt werden



## Floating-Umgebungen

- Bei `table` und `figure` handelt es sich um *floating*-Umgebungen
- sie werden von  $\text{\LaTeX}$  automatisch positioniert, erscheinen also nicht notwendigerweise an der Stelle, an der sie im Quelltext stehen
- Man kann dieses Verhalten jedoch beeinflussen, indem man am Ende des jeweiligen `begin`-Befehls eine oder mehrere der folgenden Optionen anhängt:

`h` here (Selbe Position wie im Quellcode)

`t` top (Anfang der nächsten Seite)

`b` bottom (am Seitenende)

`p` page (auf eigener Seite)

Biespiel:

```
\begin{figure}[h]
```

## Querverweise

- Man kann verschiedenen Umgebungen ein `label` hinzufügen, mit dem diese Umgebung referenziert werden kann

```
\begin{figure}
  \includegraphics{bilddatei}
  \caption{Ein schönes Bild}
  \label{abb:schoenesb}
\end{figure}
[...]
```

Abbildung `\ref{abb:schoenesb}` ist ein schönes Bild.

- Das Label darf erst nach der `\caption` stehen, da es sich sonst nicht auf die Abbinungsnummer bezieht.

## Strukturierung – Demo

## Strukturierung der Datei

Manchmal ist es sinnvoll die `.tex`-Datei in mehrere Dateien aufzuteilen, hierfür sieht  $\text{\LaTeX}$  zwei Befehle vor:

- `\input{datei}`, Inhalt von `datei` wird einfach in die aufrufende Datei übernommen
- `\include{datei}`, wie oben nur mit einem davor eingefügten Seitenumbruch

```
\section{Teil 1}  
\input{teil1}
```

```
\section{Teil 2}  
\input{teil2}
```

Abbildung: Strukturierung mit `input`

- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen
- 5 Formatierungen
- 6 Strukturierung
- 7 Mathematikmodus**
- 8 Zusammenfassung

# Mathematikmodus I

- Um mathematische Formeln zu setzen, muss man in den **Mathemodus** wechseln
- Im Mathemodus funktioniert vieles anders, und manche Zeichen haben eine andere Bedeutung
- Mathemodus wird mit `$` und `$` oder `\begin{math}` und `\end{math}` begrenzt
- Formeln bzw. Gleichungen werden auf eigene Zeilen gesetzt mit `\begin{displaymath}` bzw. `\begin{equation}`

$$a^2 + b^2 = c^2$$

$$a^2 + b^2 = c^2$$

$$c = \pm\sqrt{a^2 + b^2}$$

$$c = \pm\sqrt{a^2 + b^2}$$

$$\sum_{i=1}^n i = \frac{n \cdot (n+1)}{2}$$

$$\sum_{i=1}^n i = \frac{n \cdot (n+1)}{2}$$

## Mathematikmodus II

L<sup>A</sup>T<sub>E</sub>X bietet enorm viele Möglichkeiten Mathematik aufzuschreiben...

Eine graphische Übersicht gibt es auch hier:

<http://www.codecogs.com/latex/eqneditor.php>

$$\sum_1^n, \prod_1^n, \bigcup_{a \in A}$$

$$\vec{a} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, M = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}$$

$$y = \frac{a}{b+1} = \frac{a}{b+1}$$

$$f(x) = \int_1^\infty \frac{1}{x^2} dx = 1$$

a)	$y =$	$c$	$(konstant)$	} Polynome
b)	$y =$	$cx + d$	$(linear)$	
c)	$y =$	$bx^2 + cx + d$	$(quadr.)$	
d)	$y =$	$ax^3 + bx^2 + cx + d$	$(kubisch)$	

## Mathemodus – Demo



- Text in fester Laufweite lässt sich mit `\texttt{...}`<sup>6</sup> erzeugen.
- Innerhalb der `verbatim`-Umgebung werden alle Zeichen wörtlich interpretiert (auch  $\LaTeX$ -Steuerzeichen). Sie kann auf 2 Arten aufgerufen werden:
  - ① `\verb|...|` wenn der Text innerhalb einer Zeile steht
  - ② `\begin{verbatim} ...`
- Eine besonders schöne Möglichkeit bietet das Paket `listings`, damit sind u.A. automatische Formatierung und Zeilennummerierungen möglich.

---

<sup>6</sup>Ältere Entsprechung: `{\tt ...}`

## Paket listings

```
1  /* Ein Kommentar */
2  #include <iostream>
3
4  int main (int argc, char * const argv[]) {
5      // und noch einer!
6      std::cout << "Hello, \u0026World!\n";
7      return 0;
8  }
```

Die Darstellung kann mittels `\lstset{}` sehr fein eingestellt werden, das Paket bietet auch vordefinierte Einstellungen für verschiedene Programmiersprachen (im Beispiel C++)

- 1 Einführung
- 2 Arbeiten mit L<sup>A</sup>T<sub>E</sub>X
- 3 Spezielle Pakete
- 4 Umgebungen
- 5 Formatierungen
- 6 Strukturierung
- 7 Mathematikmodus
- 8 Zusammenfassung**

# L<sup>A</sup>T<sub>E</sub>X – Zusammenfassung I

- L<sup>A</sup>T<sub>E</sub>X ist eine Auszeichnungssprache für das Textsatzsystem T<sub>E</sub>X
- L<sup>A</sup>T<sub>E</sub>X-Dateien sind *plain text* und werden als .tex-Dateien abgespeichert. Daraus können dann verschiedene Formate generiert werden
- In der ersten Zeile eines L<sup>A</sup>T<sub>E</sub>X-Dokuments muss eine Dokumentklasse angegeben werden
- zusätzliche Pakete, Metadaten sowie neu definierte Befehle können in der Präambel angegeben werden
- Der Inhalt des Dokuments befindet sich in der Dokumentumgebung
- Textformatierung erfolgt mit Hilfe von Befehlen
- Für besonders formatierte Bereiche (Tabellen, Listen etc.) verwendet man weitere Umgebungen innerhalb des Dokuments

## L<sup>A</sup>T<sub>E</sub>X – Zusammenfassung II

- Mathematische Formeln erfordern den Mathemodus
- Jede Dokumentklasse stellt eine bestimmte Kapitel-/Abschnittstruktur zur Verfügung, anhand dieser Struktur kann automatisch ein Inhaltsverzeichnis generiert werden
- Gibt man einem Abschnitt, einer Tabelle oder einer Abbildung ein `\label{}`, so kann an anderen Stellen mit `\ref{}` wieder darauf verwiesen werden
- zusätzliche Pakete bieten zusätzliche Optionen. Sie werden in der Präambel mit `\usepackage[options]{Paketname}` geladen.
- Pakete kann man von CTAN herunterladen