

# L<sup>A</sup>T<sub>E</sub>X in 42 Schritten

Hans Peter Wolf

20. November 2006

# Historisches – Entstehung

- → Idee von D. E. Knuth (1. Februar 1977):  
für *The Art of Computer Programming*  
hochwertiger Textsatz mittels Laser-Drucker
  - Einzelpunkte sind adressierbar
  - beliebige Linien realisierbar
- → Vorbereitung:  
Gespräche mit Setzern und Druckfachleuten  
Regeln zur Konstruktion von Buchstaben und Texten
  - Dokument-, Seiten-, Abschnittsaufbau
  - explizite gegenstandsbezogene Befehle
  - Buchstaben / Wörter:  
Proportionalschrift / Kerning / Ligaturen:  
Waffel → Waffel → Waffel → Waffel  
Treffer → Treffer → Treffer → Treffer

## Historisches – Entwicklung

- Entwicklungsstufen:
  - bis Mai 1982: T<sub>E</sub>X78
  - bis 1985: T<sub>E</sub>X82
  - bis 1991: Fehlerbehebungen
  - dannach: (quasi) fehlerfrei
- Leslie Lamport (1985) — L<sup>A</sup>T<sub>E</sub>X:
  - Makro-Paket mit mächtigen, vereinfachten Befehlen
  - auch für Nicht-Informatiker
- inzwischen viele Zusatzpakete für alle möglichen Detailwünsche

# Gestaltung durch Befehle

- Gestaltung: über Befehle und Formatierungsanweisungen, genauer Makros: `\newpage` erzeugt bspw. Seitenvorschub
- Zeichen `\` zeigt einen Befehlsbeginn an
- Befehlsname auf `\`-Zeichen (meist sehr sprechend)
- Argumente von Makros folgen in geschweiften Klammern: `{...}`
- Kommentarzeichen: `%`  
alle Zeichen hinter dem Prozentzeichen werden übergangen
- Beispiele:

```
\begin{center}  
  \textbf{zentrierter} Text  
\end{center}
```

*zentrierter Text* wird zentriert und *zentrierter* wird fett gedruckt:

**zentrierter** Text

# Schritte der Dokument-Verarbeitung

1. schreibe Textdatei, bestehend aus Formulierungen und Formatierungsanweisungen: `.tex`-Datei
  2. formatiere diese Datei mit dem Textformatierer  $\text{\LaTeX}$  zu einer `.dvi`-Datei
  3. betrachte Formatierungsergebnis mit Viewer:  
`yap`, `xdvi`
  4. oder erstelle eine `.ps` bzw. `.pdf`-Datei, die sich ausdrucken lässt
- in einer Datei mit Endung `.log` legt  $\text{\LaTeX}$  alle Verarbeitungsmeldungen ab
- während der Formatierung legt  $\text{\LaTeX}$  eine `.aux`-Datei für Querverweise an (→ Inhaltsverzeichnis und Seitenreferenzen)
- !!! mit welchem Instrument (Editor) eine `.tex`-Datei erstellt wird, ist egal, jedoch erlauben manche Instrumente die Prozesse 2., 3. und 4. zu starten

# Dokumentstruktur

- Präambel (Vorspann) für Seitenlayout
  - Wahl einer Dokumentklasse  $\in$  {book, article, letter, slides, ...}  
z.B.: `\documentclass{article}`
  - Größenfestlegungen, Zusatzpakete, Marko-Definitionen
- Dokument selbst, eingeschlossen in:  
`\begin{document} ... \end{document}`
- Beispiel für Dokumentklasse `article` und Zusatzpaket `ngerman` für neue deutsche Rechtschreibung:

```
\documentclass{article}
\usepackage{ngerma}
\begin{document}
  Mein erster Text: Hallo Welt!
\end{document}
```

# Zeichen und ihre Erzeugung

- jedes Zeichen gehört zu einer von 16 Zeichenkategorien (`catcode`)
  - `\` ∈ Kategorie 0 der Escape-Zeichen
  - `{` ∈ Kategorie 1 der Gruppen-Beginn-Zeichen
  - `%` ∈ Kategorie 14 der Kommentar-Zeichen
  - Buchstaben ∈ Kategorie 11 der Buchstaben `A,...,Z,a,...z`
  - Ziffern ∈ Kategorie 12 der sonstigen Zeichen
  - ...

- alle Zeichen, die nicht in die Buchstabenklasse gehören, müssen erzeugt werden, oft durch Makro der Form `\Zeichen`, wie:

|          |                 |                  |                     |                 |                 |                 |                 |                  |
|----------|-----------------|------------------|---------------------|-----------------|-----------------|-----------------|-----------------|------------------|
| zu tun:  | <code>\#</code> | <code>\\$</code> | <code>\&amp;</code> | <code>\_</code> | <code>\%</code> | <code>\{</code> | <code>\}</code> | <code>\\$</code> |
| Wirkung: | <code>#</code>  | <code>\$</code>  | <code>&amp;</code>  | <code>_</code>  | <code>%</code>  | <code>{</code>  | <code>}</code>  | <code>§</code>   |

- das Zeichen `\` lässt sich erzeugen durch `$$\backslash$`, `\verb+\+` oder mit vielen Zeichensätzen mit `\char'134`
- das Zeichen `~` können wir schreiben mittels `\char'176` oder `\~{}`

# Akzente und Umlaute

- Akzente werden durch Makros erzeugt:

zu tun `\'o` `\'o` `\^o` `\~o` `\=o` `\.o` `\u o`  
Wirkung: ò ó ô õ ö ÷

zu tun: `\v o` `\H o` `\t{oo}` `\c o` `\d o` `\b o`  
Wirkung: ǒ Ǔ ō ȝ Ȟ ȟ

- Umlaute werden *zuverlässig* durch Makros erzeugt

zu tun: `\"a` `\"o` `\"u` `\"A` `\"O` `\"U` `{\ss}`  
Wirkung: ä ö ü Ä Ö Ü ß

!!! der Befehl `\input{uml-ok.tex}` in der Präambel ermöglicht (fast) immer, Umlaute als Umlaute einzutippen\*

\*`uml-ok.tex` ist eine Datei, die durch den `input-`Befehl eingelesen wird. Ohne Pfad muss diese Datei im Verzeichnis der `.tex`-Datei stehen. Bezug: [www.wiwi.uni-bielefeld.de/~wolf/software/tex/uml-ok.tex](http://www.wiwi.uni-bielefeld.de/~wolf/software/tex/uml-ok.tex)



# Dimensionen des Umlautproblems

- zu unterscheiden sind
  - das Zeichen, das auf einer Taste steht
  - der Code, der nach Tastendruck in einem Editor ankommt
  - das Zeichen, das am Bildschirm dargestellt wird
  - das Zeichen, das nach Speicherung in einer Datei ankommt
  - das Zeichen, das  $\text{\LaTeX}$  einem gelesenen Zeichen zuordnet
  - das Zeichen, das sich an der adressierten Position im Zeichensatz befindet ...

... es kommen im Fehlerfall also mehrere Ursachen infrage

- Hinweis1: werden besondere Zeichen durch Makros generiert, tauchen keine Umlautprobleme auf
- Hinweis 2: der Trennalgorithmus kann leider keine besonderen Zeichen wie Umlaute verarbeiten

## ad Umlautproblem: 2 Pakete

- speichert der verwendete Editor ein ä gemäß Latin1, dann findet  $\text{\LaTeX}$  mit den Setzungen `\usepackage[T1]{fontenc}` und `\usepackage[latin1]{inputenc}` aus der Tabelle der DC-Fonts das richtige Zeichen, so es vorhanden ist
  - mit `\usepackage[latin1]{inputenc}` (Input-Encoding) interpretiert  $\text{\LaTeX}$  Zeichen aus Dateien nach dem Latin1-Standard
    - die Oktalwerte 344, 366, 374, 304, 326, 334 und 337 entsprechen ä, ö, ü, Ä, Ö, Ü und ß
    - Latin1 entspricht ISO-8859-1 (West European languages)
  - mit `\usepackage[T1]{fontenc}` sucht  $\text{\LaTeX}$  die Zeichen in der Font-Tabelle nach der 1990 in Cork beschlossenen Zuordnung
    - die druckbaren der ersten 128 Zeichen stehen in der Tabelle an derselben Stelle wie im ascii-Zeichensatz
    - ä, ö, ü, Ä, Ö, Ü, ß stehen an den Oktal-Positionen 344, 366, 374, 304, 326, 334 und 377, wogegen oktal 337 ein „SS“ liefert
- !!! Inkompatibilitäten zusätzlich geladener Pakete oder unzureichende Zeichensätze können leider wieder Fehler hervorrufen

## Das Paket `ngerman`

- stellt deutsche Überschriften bereit wie:  
Table of contents → Inhaltsverzeichnis
- Makros für deutsche Anführungszeichen:  
`\glqq Hallo Welt!\grqq` → „Hallo Welt!“
- deutsche Trenntabelle
- das `ngerman`-Paket erlaubt Umlaute zu schreiben durch:

|          |    |    |    |    |    |    |    |
|----------|----|----|----|----|----|----|----|
| zu tun:  | "a | "o | "u | "A | "O | "U | "s |
| Wirkung: | ä  | ö  | ü  | Ä  | Ö  | Ü  | ß  |

!!! das Zeichen " wird durch `ngerman` zu einen *active character*, was manchmal Probleme bereiten kann – dieses lässt sich beheben durch `\catcode' "=12`

# Gliederung von Dokumenten

- in der Klasse `article` lassen sich Dokumente mit den Befehlen `\section`, `\subsection`, `\subsubsection` und `\paragraph` gliedern
- die Klasse `book` kennt daneben noch `\chapter` und `\part`
- den Gliederungsbefehlen sind die gewünschten Überschriften als Parameter zu übergeben
- Beispiel für eine Gliederungsstruktur:

```
\section{Einleitung}
\subsection{Das Problem}
\subsection{Die Lösungsstrategie}
\subsection{Weg der Beschreibung}
\section{Das Problem des Problems}
\subsection{Historische Bemerkungen}
...
\section{Fazit}
```

# Abschnittsgestaltung

- eingerückt:

```
Text Text Text
\begin{quote}
  eingerückter \\ Text
\end{quote}
Text Text Text
```

|                      |
|----------------------|
| Text Text Text       |
| eingerückter<br>Text |
| Text Text Text       |

- zentriert:

```
Text Text Text
\begin{center}
  eingerückter \\ Text
\end{center}
Text Text Text
```

|                      |
|----------------------|
| Text Text Text       |
| eingerückter<br>Text |
| Text Text Text       |

!!! Hinweis: \\ erzeugt Zeilenumbruch innerhalb eines Abschnitts  
\\[3mm] fügt zusätzlich 3mm *vertikale Luft* ein

!!! Hinweis: \begin{flushleft}...\end{flushleft} verhindert Blocksatz,  
\raggedright unterdrückt Blocksatz im weiteren Dokument

# Auflistungen

- Itemize-Umgebung

```
\begin{itemize}
  \item erster Punkt
  \item zweiter Punkt
  \item dritter Punkt
    \begin{itemize}
      \item U-Punkt 1
      \item U-Punkt 2
    \end{itemize}
  \item[xx] noch einer
\end{itemize}
```

|  |
|--|
| <ul style="list-style-type: none"><li>• erster Punkt</li><li>• zweiter Punkt</li><li>• dritter Punkt<ul style="list-style-type: none"><li>– U-Punkt 1</li><li>– U-Punkt 2</li></ul></li></ul> <p>xx noch einer</p> |
|--|

!!! Hinweis: mittels des optionalen Argumentes von `\item` im Beispiel `xxx` in eckigen Klammern wird die Markierung selbst gesetzt

# Aufzählungen

- Enumerate-Umgebung

```
\begin{enumerate}
  \item erster Punkt
  \item zweiter Punkt
  \item dritter Punkt
    \begin{enumerate}
      \item U-Punkt 1
      \item U-Punkt 2
    \end{enumerate}
  \item[999] noch einer
\end{enumerate}
```

|                  |
|------------------|
| 1. erster Punkt  |
| 2. zweiter Punkt |
| 3. dritter Punkt |
| (a) U-Punkt 1    |
| (b) U-Punkt 2    |
| 999 noch einer   |

!!! Hinweis: Itemize- und Enumerate-Umgebungen können geschachtelt werden!

!!! Aufgabe: finde heraus, was die description-Umgebung leistet

# Befehlssyntax

- Makro ohne Argumente  
`\Befehlsname`  
`\item, \maketitle, \newpage`
- Makro mit einem Argument  
`\Befehlsname{Argument}`  
`\section{Einleitung}, \usepackage{ngerman}`
- Makro mit einem optionalen Argument  
`\Befehlsname[optionales Argument]`  
`\item[999], \[2ex]`
- Makro mit einem optionalen und einem notwendigen Argument  
`\Befehlsname[optionales Argument]{Argument}`  
`\usepackage[T1]{fontenc}`
- Umgebungen  
`\begin{Umgebungsname} ...`  
`\end{Umgebungsname}`  
mit `Umgebungsname`  $\in$  {center, quote, itemize, enumerate, ...}



# Modi und Seitenaufbau

## Modi

- horizontaler Modus — Zustand innerhalb Abschnitten
- vertikaler Modus — Zustand zur Anordnung von Abschnitten
- Mathe-Modus — Zustand innerhalb von Formelbearbeitungen

## Vorstellung der Seitenkonstruktion

- Buchstaben / Zeichen werden in sie umgebende Boxen platziert
- Buchstabenboxen werden zu Wortboxen vereinigt
- Zeilen entstehen durch Aneinanderreihung von Wortboxen
- aus Zeilen werden Abschnitte konstruiert
- dabei werden ggf. Trennungen gesucht
- Abschnittsboxen werden im *vertikalen Modus* vertikal zusammengefügt
- Formeln werden nach besonderen Regeln im *Mathe-Modus* gesetzt

# Schriften

- Schriftgrößen

|                |                            |
|----------------|----------------------------|
| winzig         | <code>\tiny</code>         |
| sehr klein     | <code>\footnotesize</code> |
| klein          | <code>\small</code>        |
| normal         | <code>\normalsize</code>   |
| groß           | <code>\large</code>        |
| sehr groß      | <code>\Large</code>        |
| sehr sehr groß | <code>\LARGE</code>        |
| riesig         | <code>\huge</code>         |
| sehr riesig    | <code>\Huge</code>         |

!!! diese Befehle funktionieren wie ein Schalter, Rückstellung durch `\normalsize` oder Einsatz in geschweiften Klammern

Beispiel: `\LARGE` groß `\normalsize` und `{\tiny` winzig wird erzeugt durch:

```
\fbox{\LARGE groß \normalsize und {\tiny winzig}}
```

# Schriftarten

- Schriftarten:

|                     |                                  |                |
|---------------------|----------------------------------|----------------|
| Roman-Schrift       | <code>\rm Roman</code>           | Roman          |
| fette Schrift       | <code>\textbf{bold}</code>       | <b>bold</b>    |
| Typewriterschrift   | <code>\texttt{typewriter}</code> | typewriter     |
| Italic-Schrift      | <code>\textit{italic}</code>     | <i>italic</i>  |
| serifenlose Schrift | <code>\textsf{sans serif}</code> | sans serif     |
| slanted Schrift     | <code>{\sl slanted}</code>       | <i>slanted</i> |

!!! Hinweis: unterstreichungen – wenn vermeidbar – nie!

!!! Hinweis: betone *immer* mittels `\emph`, also:

betone `\emph{immer}` mittels `\verb+\emph+`

## Längen und Maße

- Höhe, Breite, Ränder usw. können neu gesetzt werden
- einige Längeneinheiten von  $\text{\LaTeX}$ : mm, pt, in (1 inch = 25.4 mm = 72.27 pt), em (1em = Breite eines  $m$ ), ex (1ex = Höhe eines  $x$ )
- einige Befehle für die Präambel zur Layout-Veränderung:

|                        |                                 |
|------------------------|---------------------------------|
| Absatzabstand          | <code>\parskip=2ex</code>       |
| Absatzeinzug           | <code>\parindent=0mm</code>     |
| negativer Absatzeinzug | <code>\parindent=-10mm</code>   |
| Textbreite             | <code>\textwidth=14cm</code>    |
| Texthöhe               | <code>\textheight=22cm</code>   |
| linker Rand            | <code>\oddsidemargin=0mm</code> |
| oberer Rand            | <code>\topmargin=0mm</code>     |

!!! verändere Standardformatierungen nur in begründeten Fällen!

# Luft / Platz

horizontale Luft:

- 5mm: `\hspace*{5mm}`
- 5mm zurück: `\hspace*{-5mm}`
- Leerzeichen: `\quad`, `\,` , `\!`
- 2em, 1em,  $\frac{3}{8}$ em,  $-\frac{3}{8}$ em Luft: `\quad`, `\,` , `\!`

vertikaler Platz:

- 5mm vertikale vor: `\vspace*{5mm}`
- 5mm vertikale zurück: `\vspace*{-5mm}`
- ein wenig vertikale Platz: `\smallskip`
- etwas vertikale Platz: `\medskip`
- viel vertikale Platz: `\bigskip`

# Boxen

- eine Box ist eine nicht trennbare Einheit
- Links-Rechts-Box ohne Rahmen: `\mbox{TEXT}`, `\makebox[Breite][Position]{beliebiger Text}`
- Links-Rechts-Box mit Rahmen: `\fbox{TEXT}`, `\framebox[Breite][Position]{beliebiger Text}`
- mit *Position* ∈ {l,r,c} für Inhalt wird linksbündig, rechtsbündig, zentriert in Box positioniert
- Beispiele:

`X\mbox{Hallo}X` → XHalloX, `X\fbox{Hallo}X` → XHalloX

`X\framebox[6em][l]{Hallo}X` → XHalloX

`X\makebox[6em][r]{Hallo}X` → XHalloX



`X\makebox[6em][c]{Hallo}X` → XHalloX

`X\framebox[1em][c]{Hallo}X` → HalloX

!!! vertikale verschobene Boxen:

`xxx\raisebox{1ex}{xxx}xxx` → xxx<sup>xxx</sup>xxx

## Schwarze Boxen

- *rule*-Boxen-Befehl: `\rule[vertikale Verschiebung]{\rule[vertikale Verschiebung]{Breite}{Höhe}}`
- schwarze Rechtecke: `\rule[3mm]{4mm}{2mm}` → 
- horizontale Linien: `\rule{20mm}{.1mm}` → \_\_\_\_\_
- vertikale Linien: `\rule{.1mm}{4mm}` → |
- gerahmte Box: `\fbox{\rule[1mm]{3mm}{3mm}}` → 

# Tabellen

- Gerüst einer Tabelle

```
\begin{tabular}{\langle Spaltendesign \rangle}
\langle Body der Tabelle \rangle
\end{tabular}.
```

- für  $\langle Spaltendesign \rangle$  können eingesetzt werden l, c, r und p{...}

```
\begin{tabular}{rcl}
\hline\[-1ex]
Nr.& Symbol & Bedeutung
\[/.5ex]\hline\[-1.5ex]
1& l & & linksbündig\\
2& c & & zentriert\\
3& r & & rechtsbündig\\
4& \verb+p{20mm}+ & & 20-mm-Spalte\\
5& \verb-& & Trennzeichen\\
6& \verb-\\- & & Zeilenumbruch\\
7& \verb+\hline+ & & horizontale Linie
\[/.5ex]\hline\[-1.5ex]
\end{tabular}
```

| Nr. | Symbol  | Bedeutung         |
|-----|---------|-------------------|
| 1   | l       | linksbündig       |
| 2   | c       | zentriert         |
| 3   | r       | rechtsbündig      |
| 4   | p{20mm} | 20-mm-Spalte      |
| 5   | &       | Trennzeichen      |
| 6   | \\      | Zeilenumbruch     |
| 7   | \hline  | horizontale Linie |



# Tabellen: Spaltendesignverfeinerungen

- Sprachelemente zum Spaltendesign:

|    |                 |   |   |
|----|-----------------|---|---|
| 1. |                 | : | plaziert vertikale Linie                            |
| 2. | @{:}            | : | plaziert zwischen Spalten:                          |
| 3. | @{\quad:\quad}  | : | plaziert zwischen Spalten: und etwas Luft           |
| 4. | @{\hspace{1cm}} | : | setzt 1cm Luft zwischen Spalten                     |
| 5. | @{}             | : | entfernt kleinen Zwischenraum zwischen zwei Spalten |

- Quellcode:

```
\begin{tabular}{||l|l@{\quad:\quad}p{6cm}||} \hline
1. & \verb-|- & & plaziert vertikale Linie & \\
2. & \verb?@{:}? & & plaziert zwischen Spalten: & \\
3. & \verb?@{\quad:\quad}? & & & \\
& & & & plaziert zwischen Spalten: und etwas Luft \\
4. & \verb?@{\hspace{1cm}}? & & & setzt 1cm Luft zwischen Spalten \\
5. & \verb?@{}? & & & entfernt kleinen Zwischenraum \\
& & & & zwischen zwei Spalten & \\
\hline \end{tabular}
```

# Tabellen-Spezialitäten

- Zusammenfassung von Spalten einer Zeile  
`\multicolumn{Spaltenzahl}{Spaltentyp}{Text}`
- Anhebung eines Eintrags mit `\raisebox`
- Quellecode:

```
\begin{tabular}{rrrrr}\hline
& & \multicolumn{3}{c}{Haarfarbe} & \\
& & blond & braun & schwarz & \\
&M& 13 & 17 & 10 & \\
\raisebox{2ex}[-2ex]{Geschlecht}
&W& 11 & 14 & 18 & \\
\hline\end{tabular}
```

- Wirkung:

|            |  | Haarfarbe |       |         |
|------------|--|-----------|-------|---------|
| Geschlecht |  | blond     | braun | schwarz |
| M          |  | 13        | 17    | 10      |
| W          |  | 11        | 14    | 18      |

# mathematische Formeln

- stelle im Text auf Mathe-Modus durch Umschalter \$ um: text \$formel\$ text
- schlieÙe abgesetzte Formeln in Klammern ein: \[ formel \]
- schreibe Formeln wie man sie spricht
- setze Text in Formeln durch \mbox{TEXT...}
- verwende Markos für griechische Buchstaben: \alpha, \beta, \pi, \mu für  $\alpha, \beta, \pi, \mu$ .
- schreibe Exponenten mit Hütchen ^:  $x^2, e^{-(x-\mu)^2/2}, x^{2^2}$  für  $x^2, e^{-(x-\mu)^2/2}, x^{2^2}$
- schreibe Indizes mit Unterstrich \_:  $x_1, y_{t-2}, z_2^3$  für  $x_1, y_{t-2}, z_2^3$
- erstelle Brüche mittels \frac{...}{...}: \frac{1}{2}, -\frac{1}{2}\frac{(x-\mu)^2}{\sigma} für  $-\frac{1}{2}, \frac{(x-\mu)^2}{\sigma}$
- schaffe Platz in Formeln durch \, \quad, \qquad

## abgesetzte Formeln

- `\begin{equation}`  
    `\int_{-\infty}^{\infty}`  
    `f(x)\, dx=1`  
`\end{equation}`

erzeugt abgesetzte Formel mit Nummerierung:

$$\int_{-\infty}^{\infty} f(x) dx = 1 \quad (1)$$

- entsprechende Formel ohne Nummerierung durch:  
`\begin{eqnarray*}... \end{eqnarray*}`
- Gestalt der Referenznummern hängt von Dokumentklasse\* ab
- Gestaltänderung Referenznummern über Re-Definition von `\theequation`

\*für class book: `\arabic{chapter}.\arabic{equation}`

# große Symbole, Funktionen und Klammern

- Summen-, Integral-, Produkt-, Grenzwert-, Wurzelzeichen:  
für

$$\sum_{i=1}^n i \quad \int_{x_0}^1 f(t) dt \quad \prod_{i=1}^n i \quad \lim_{x \rightarrow 0} \sqrt[3]{1000}$$

schreibe: `\sum_{i=1}^n i`, `\int_{x_0}^1 f(t)`, `dt`,  
`\prod_{i=1}^n i`, `\lim_{x \to 0}` bzw.  
`\sqrt[3]{1000}`

- viele Funktionen werden durch Makros erstellt:  
`exp sin cos tan cot log ln min max`  
`\exp \sin \cos \tan \cot \log \ln \min \max`
- linke Klammer für einen Term: `\leftKlammer`,  
rechte Klammer für einen Term: `\rightKlammer`  
mit  $Klammer \in \{ \{, [, (, \}, ], ) \}$   
Fallunterscheidungsklamme:
  - schreibe als (linke) Unterscheidungsklammer  
`\left\{`
  - setze rechte unsichtbare Klammer unbedingt:  
`\right.`

# Gleichungssysteme

- Gleichungssysteme sind wie an =-Zeichen ausgerichtete Tabellen:

$$z_1 = 2x_1 + 3y_2 \quad (2)$$

$$z_2 = 4x_1 - 2y_2 \quad (3)$$

— diese erhalten wir durch:

```
\begin{eqnarray}
z_1 & = & 2x_1+3y_2 \\
z_2 & = & 4x_1-2y_2
\end{eqnarray}
```

- Tabellenstrukturen in Formeln werden mit der array-Umgebung gesetzt (Syntax entspricht tabular-Umgebung); erstelle bspw.

$y = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{pmatrix}$ 
 durch:
 

```
\[ y=\left(
\begin{array}{cc}
1&2 \\ 3&4 \\
5&6 \\ 7&8
\end{array}
\right)
```

## Sonderzeichen in Formeln

- Spezialzeichen:

$\times$     $\cap$     $\cup$     $\vee$     $\wedge$     $\setminus$     $\circ$     $\diamond$   
`\times`   `\cap`   `\cup`   `\vee`   `\wedge`   `\setminus`   `\circ`   `\diamond`

- Punkte:

$\cdot$     $\cdots$     $\dots$     $\ddots$     $\vdots$     $\cdot$   
`\cdot`   `\cdots`   `\ldots`   `\ddots`   `\vdots`   `\cdot`

- Relations- und weitere Zeichen:

$<$     $\leq$     $=$     $\geq$     $>$     $\neq$     $|$     $\sim$     $\approx$     $\infty$   
`<`   `\le`   `=`   `\ge`   `>`   `\ne`   `\mid`   `\sim`   `\approx`   `\infty`

- Pfeile (setze jeweils für \* arrow ein!):

$\leftarrow$     $\rightarrow$     $\uparrow$     $\downarrow$     $\Leftrightarrow$     $\Rightarrow$     $\Uparrow$   
`\left*`   `\right*`   `\up*`   `\down*`   `\Left*`   `\Right*`   `\Up*`  
 $\Downarrow$     $\leftrightarrow$     $\nearrow$     $\searrow$     $\swarrow$     $\nwarrow$   
`\Down*`   `\leftrigh*`   `\ne*`   `\se*`   `\sw*`   `\nw*`

# Bilder

- Integration von Postscript-Bildern:

`\usepackage{graphicx}` in Präamble, Referenz mit:  
`\includegraphics[Optionen]{Dateiname}`

Beispiel: `\includegraphics{roulettebrett.ps}`



**Abb.:** Ein Roulette-Brett

- Postscript-Bilder müssen Bounding-Box enthalten
- mit pdflatex lassen sich auch JPEGs u.a. Formate einbauen



## Graphiken und figure

- Bildunterschriften mittels figure-Umgebung:

```
\begin{figure}[h]
  \includegraphics{roulettebrett.ps}
  \caption{Ein Roulette-Brett}
\end{figure}
```

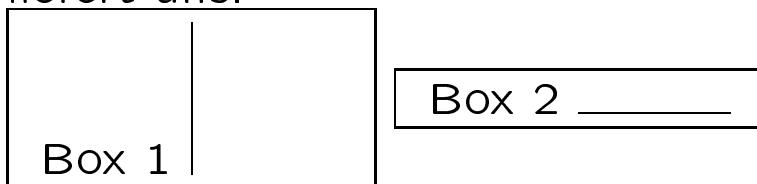
- `\caption` legt Untertitel-Text fest
- `h` empfiehlt das Bild *here* zu platzieren
- wandert das Bild trotzdem unschön nach hinten hilft:  
`\usepackage{float}` und figure-Option `H`

# Minipages

Minipages zur Platzierung von Textboxen nebeneinander:

```
\fbox{
  \begin{minipage}{4cm}
    Box 1 \rule{.1mm}{2cm}
  \end{minipage}
}
\fbox{
  \begin{minipage}{4cm}
    Box 2 \rule{2cm}{.1mm}
  \end{minipage}
}
```

liefert uns:



## eigene Strukturen

- Makro ohne Parameter:  
`\newcommand{\mymacro}{Inhalt des Makros}`
- Makro mit einem Parameter:  
`\newcommand{\myparmacro}[1]{Argument ist #1}`
- Makro mit zwei Parametern:  
`\newcommand{\withtwopars}[2]{arg1: #1, arg2: #2}`
- eigene Zähler: definieren, setzen, hochzählen:\\  
`\newcounter{mycount}`  
`\setcounter{mycount}{3}`  
`\addtocounter{mycount}{2}`  
`\stepcounter{mycount}`  
Verwendung:~~arabisch, römisch:  
`\arabic{mycount}, \roman{mycount}`  
Seite drucken:  
`\setcounter{mycount}{\value{page}}\arabic{mycount}`  
eigene Zähler: definieren, setzen, hochzählen:  
Verwendung: arabisch, römisch: 6, vi  
Seite drucken: 34

etc.

- kurze Striche: - -- --- \$-\$ → - - — -
- besondere Buchstaben:  
œ Œ æ Æ å Å ø Ø † ‡  
`\oe \OE \ae \AE \aa \AA \o \O \l \L`
- Randnotizen:  
`\marginpar{hallo}`
- Fußnoten: `\footnote{vermeide Fußnoten!}`\*
- Verweise:  
Marke setzen mit `\label{stellxyz}`,  
Seite referenzieren mit `\pageref{stellxyz}`,  
Einheit referenzieren mit `\ref{stellxyz}`
- Input von ganzen T<sub>E</sub>X-Dateien:  
`\input{Dateiname}`
- Inhaltsverzeichnis durch Befehl `\tableofcontents`

\*vermeide Fußnoten!

## Warnungen und Fehler

- häufige Warnung:  
`Overfull hbox ...: LATEX` konnte keine schönen Trennungen finden und hat über den Rand geschrieben  
→ baue Trennhilfen ein: \-
- noch eine häufige:  
`Underfull hbox ...: LATEX` konnte keine schönen Trennungen finden und es ist zu viel Luft in den Zeilen  
→ baue Trennhilfen ein: \-
- Fehlermeldungen:  
ein Kapitel für sich; lese in Ruhe die Meldung, betrachte die angezeigten Zeile / suche im Quelltext die Zeile über die Zeilennummer

## Literaturverzeichnisse

```
\begin{thebibliography}{xyz}
\bibitem[Knu73]{DEK73} Donald E.\ Knuth, 1973,
    \textit{The Art of Computer Programming}
\bibitem[Kuerzel2]{Code2} Autor2, Jahr2, Titel2,...
\end{thebibliography}
```

führt zu dem kurzen Verzeichnis:

```
[Knu73]    Donald E. Knuth, 1973, The Art of
           Computer Programming
[Kuerzel2] Autor2, Jahr2, Titel2, ...
```

mit `\cite{DEK73}` fordern wir im Text einen Verweis an und erhalten dann [Knu73] an der Stelle eingetragen

Hinweis: fehlt das optionale Argument hinter `\bibitem`, wird als Kürzel der Code verwendet.

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

- L<sup>A</sup>T<sub>E</sub>X-Shells zur Vereinfachung der Arbeit  
... gibt es viele, siehe:

<http://www.itsfd.de/texwin/index.php?site=vcat&cat=33&lang=de>

- Texmaker, siehe:

<http://www.xmlmath.net/texmaker>

wunderbar und läuft unter Windows, Linux und MacOSX

- TeXnicCenter, siehe:

<http://www.toolscenter.org/>

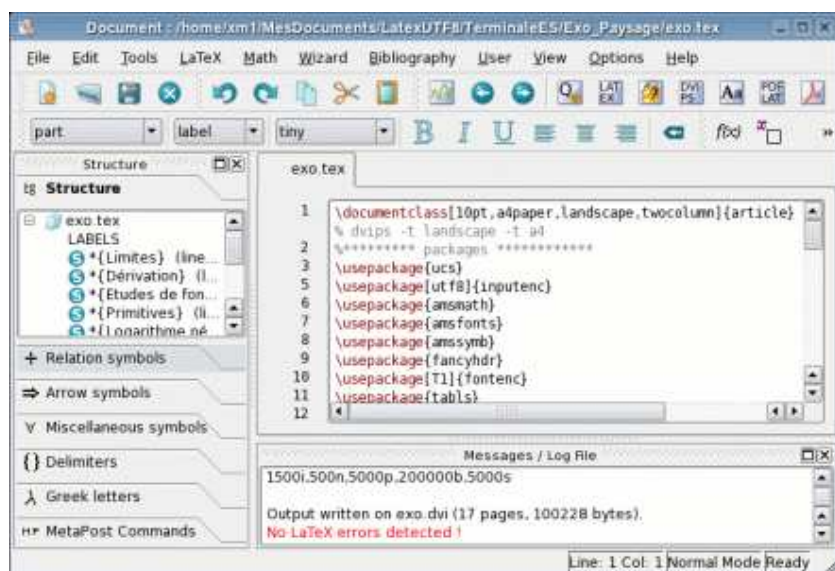
kostet nichts kostet und wird gern empfohlen

- WinEdt, siehe:

<http://www.winedt.com/>

verbreitet und beliebt, jedoch leider nur shareware

# Texmaker – Screen





# Folien

- Folien (einfachst):

```
\documentclass{slides}
...
\begin{document}
  \begin{frame}[fragile]
    Folientext
  \end{frame}
  \begin{frame}[fragile]
    ...
  \end{frame}
\end{document}
```

- moderner Dokumentklasse seminar
- beamer-Dokumentklasse, siehe bspw.:

```
http://mo.mathematik.uni-stuttgart.de/
kurse/kurs44/abschnitt1_4.html
```

Hinweis: nicht alle Elemente zusätzliche ladbarer Pakete harmonieren!

## Literatur, kaufbar

- [1] M. Goosens, F. Mittelbach, A. Samarin: *Der L<sup>A</sup>T<sub>E</sub>X-Begleiter*, Addison-Wesley
- [2] H. Kopka: *LaTeX: Band 1 - Eine Einführung*, Addison-Wesley
- [3] H. Kopka: *LaTeX: Band 2 - Ergänzungen*, Addison-Wesley
- [4] H. Kopka: *LaTeX: Band 3 - Erweiterungen*, Addison-Wesley

... erschienen in verschiedenen Auflagen

# Links

- The Comprehensive TeX Archive Network:  
<http://www.ctan.org/>
- wesentlicher ftp-Server für T<sub>E</sub>X:  
<ftp://ftp.dante.de/pub/tex/>
- L<sup>A</sup>T<sub>E</sub>X-Befehlsübersicht im Netz:  
<http://www.weinelt.de/latex/>
- Das L<sup>A</sup>T<sub>E</sub>X-Sündenregister – sehr zu empfehlen:  
<http://www.ctan.org/tex-archive/info/l2tabu/german/l2tabu.pdf>
- eine Übersicht über L<sup>A</sup>T<sub>E</sub>X-Editoren:  
<http://www.itsfd.de/texwin/index.php?site=vcat&cat=33&lang=de>
- deutsche L<sup>A</sup>T<sub>E</sub>X-Kurzbeschreibung:  
<ftp://ftp.dante.de/pub/tex/info/lshort/german/l2kurz.pdf>
- Seite des Texmaker:  
<http://www.xmlmath.net/texmaker>
- Seite des T<sub>E</sub>XnicCenter:  
<http://www.toolscenter.org/>
- Seite des winEdt:  
<http://www.winedt.com/>
- L<sup>A</sup>T<sub>E</sub>X in 14 Minuten:  
<http://www.wiwi.uni-bielefeld.de/~wolf/software/tex/vierzehn.pdf>