

LaTeX-Praxis

# Einführung

Jörn Clausen

joernc@gmail.com

# Übersicht

- Was ist  $\LaTeX$ , was ist  $\LaTeX$  nicht?
- Wie funktioniert  $\LaTeX$ ? Wieso gerade so?
- erste Schritte mit  $\LaTeX$
- Dokumente mit  $\LaTeX$  strukturieren

# Was ist $\text{\LaTeX}$ ?

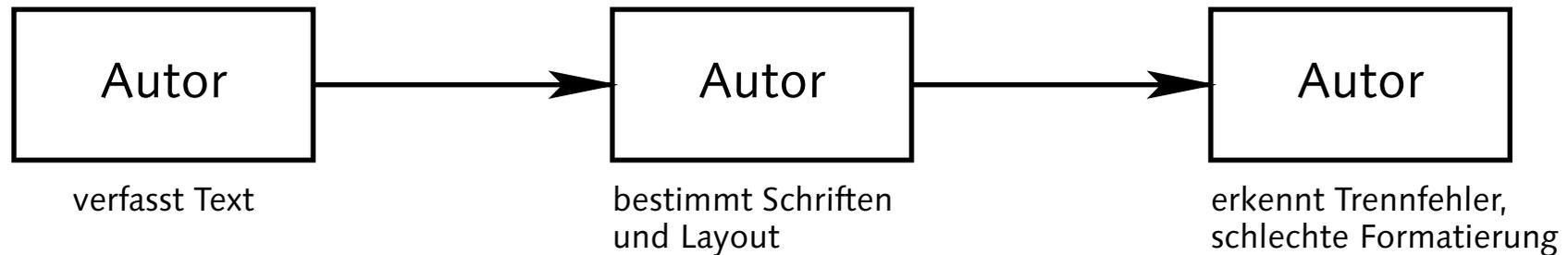
- $\text{\LaTeX}$  ...
  - ist keine Textverarbeitung, sondern ein Satzsystem
  - bietet kein WYSIWYG (what you see is what you get)
  - ist ziemlich alt
- Sind das Nachteile? Nicht wirklich:  $\text{\LaTeX}$  ...
  - liefert hochwertige Ergebnisse
  - wird von vielen Autoren und Verlagen verwendet
  - bietet Lösungen zu vielen Satz-Problemen
  - läuft auf fast jedem Computer

# Textverarbeitung – damals . . .



- Tätigkeit von Lektor/Layouter und Setzer
  - nicht trivial
  - Handwerk
  - Kunst
  - braucht Erfahrung

## ... und heute



- klassische Textverarbeitungen überfordern den Benutzer
- noch schlimmer: DTP-Programme
- Beweis: *Glückwunschkarten* mit **20 verschiedenen** SCHRIFTARTEN

# Revival eines Paradigmas

- Autor bestimmt Inhalt und logische Struktur des Textes
- „Experte“ legt Layout und Formatierung fest
- Satzsystem realisiert Ausgabe
- Dokumentklassen (Buch, Report, Brief, Kochrezept, ...)
- Autor beschreibt Bedeutung, nicht Aussehen:

hervorheben

statt

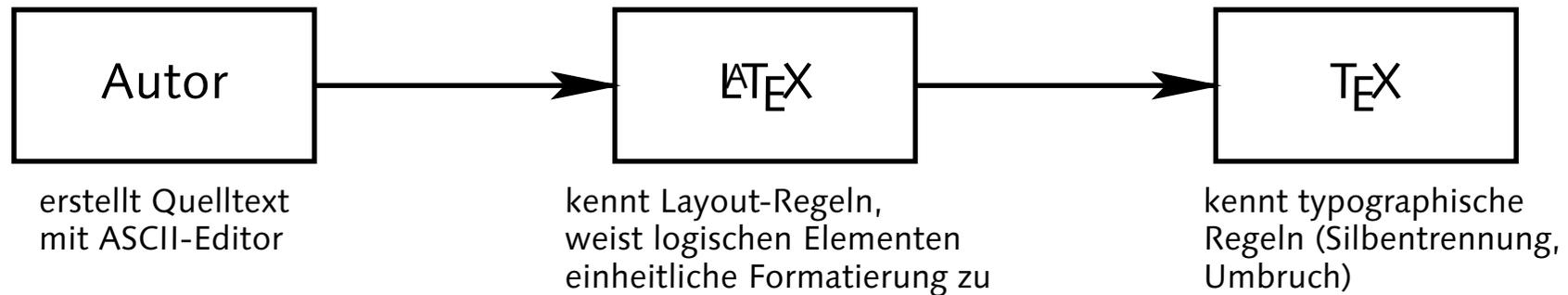
kursiv

Überschrift

statt

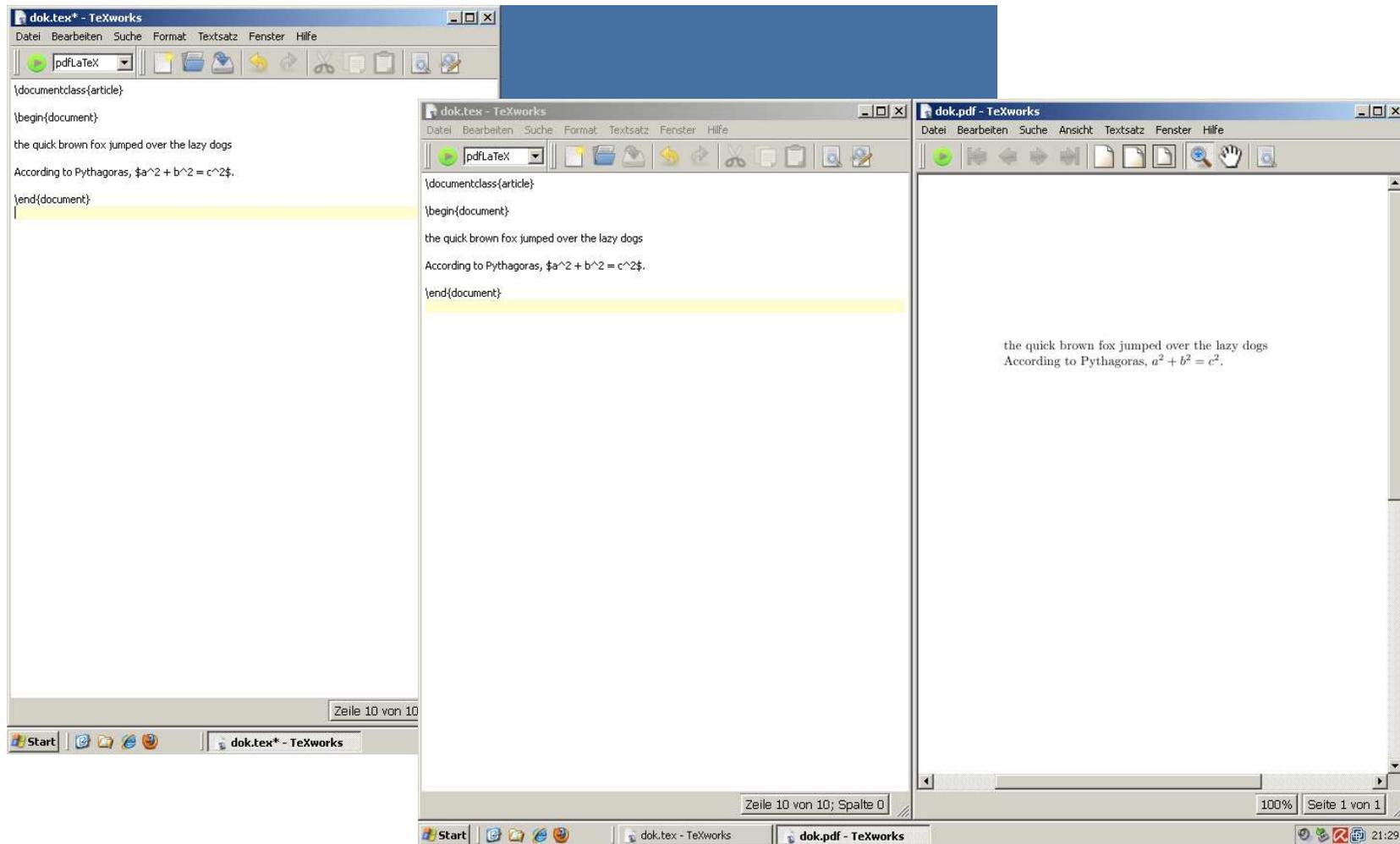
15pt, fett, zentriert

# Schriftsatz mit $\text{\LaTeX}$



- $\text{\TeX}$  1977 von Donald Knuth entwickelt
- mathematischer Schriftsatz, gut lesbare Fonts
- $\text{\LaTeX}$  1985 von Leslie Lamport entwickelt
- „frontend“ für  $\text{\TeX}$  mit mächtigen Makro-Befehlen

# L<sup>A</sup>T<sub>E</sub>X im Fenster: TeXworks



# Aufgaben

- Eine minimale  $\text{\LaTeX}$ -Datei könnte so aussehen:

```
\documentclass{article}
\begin{document}
Hello world!
\end{document}
```

Speichere sie als „hello.tex“, übersetze sie und sieh Dir das Ergebnis an. Was fällt Dir auf?

- Was könnten die einzelnen Zeilen in der Datei bedeuten?

# Befehle

- Befehle werden mit einem backslash (\) eingeleitet

- einfacher Befehl:

```
\befehl1
```

- Befehl mit einem Pflicht-Argument:

```
\befehl2{argument}
```

- Befehl mit zwei Pflicht-Argumenten:

```
\befehl3{argument1}{argument2}
```

- Befehl mit optionalem Argument:

```
\befehl2[optargument]{argument}
```

# Aufgaben

- Ersetze die erste Zeile durch

```
\documentclass[a4paper,12pt]{article}
```

und übersetze die Datei erneut. Was hat sich geändert?

- Ergänze die Hello-world-Zeile um

```
Hello world! It's \emph{so} great to use \LaTeX.
```

und übersetze das Dokument erneut.

- Welche Dateien entstehen bei der Übersetzung?

# Fehler passieren

- mehr oder weniger hilfreiche Fehlermeldungen
- nicht immer leicht zu verstehen, aber ...
- **LESEN HILFT OFT WEITER!!!**

```
! Undefined control sequence.
```

```
1.6 Hello world! It's \enph
```

```
{so} great to use \LaTeX.
```

- Übersetzung mit „x“ beenden
- Zeilennummer beachten
- Fehler direkt vor „Zeilenwechsel“

# Aufgaben

- Füge einen längeren Text in die Datei ein.

Falls Du nicht weißt, was Du schreiben sollst, kannst Du mit dem „Lipsum generator“ beliebig viel lateinisch anmutenden Text erzeugen:

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

- Welche Bedeutung haben Zeilenumbrüche im Quelltext?
- Welche Bedeutung haben Leerzeilen im Quelltext?

# Klassen und Stile

- die Dokumentklasse bestimmt Art und damit Aussehen (Layout) des Dokuments:

- kurzer Artikel

- ```
\documentclass{article}
```

- Vortragsfolien

- ```
\documentclass{slides}
```

- Stile bestimmen Art und Aussehen einzelner Bestandteile oder Aspekte des Dokuments
- Stile werden in der Präambel der  $\LaTeX$ -Datei eingebunden

- ```
\documentclass[a4paper]{article}
```

- ```
\usepackage{color}
```

- ```
\usepackage{chess}
```

- Befehle werden in Dokumentklassen und Stilen definiert

# Aufgaben

- Füge die folgende Zeile in der Präambel von `hello.tex` ein:

```
\usepackage{palatino}
```

Was ändert sich?

- Binde das Paket „`pifont`“ ein und füge diese beiden Zeilen zum Dokument hinzu:

```
\ding{41} joernc (at) gmail (dot) com
```

```
\ding{37} 0521/106-12601
```

- Binde das Paket „`createsudoku`“ ein und verwende den Befehl `\generategrid`

# Umlaute und Anführungszeichen

```
\documentclass[12pt,a4paper]{article}
\usepackage[german]{babel}
\usepackage[utf8]{inputenc}
```

```
\begin{document}
```

```
Umlaute: \"a \"o \"u \"A \"O \"U \ss{}
```

```
Akzente: \'a `a ^a \c{C} \O{} \~n
```

```
Anf\"uhrungszeichen: ``Ha!'' `Hu?'
```

```
mit babel: "a "o "u "A "O "U "s "`He!""'
```

```
mit inputenc: ä ö ü Ä Ö Ü ß á à â ø ñ
```

```
\end{document}
```

# Aufgaben

- Füge die Anweisung

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

in die  $\LaTeX$ -Datei ein. Wie ändert sich die Formatierung?

- Füge Umlaute, Akzente und Anführungszeichen in den Text ein.

# Dokumentenstruktur

```
\documentclass[12pt,a4paper]{article}

\begin{document}
  \section{Einleitung}
    \subsection{Motivation}
      aaa bbb ccc ddd eee
    \subsection{Danksagungen}
      aaa bbb ccc ddd eee
  \section{Hauptteil}
    \subsection{Andere Verfahren}
      aaa bbb ccc ddd eee
    \subsection{Unsere L\"osung}
      aaa bbb ccc ddd eee
      \subsubsection{Ansatz}
      \subsubsection{Probleme}
\end{document}
```

# Aufgaben

- Strukturiere das Dokument mit Hilfe der Befehle `\section`, `\subsection` und `\subsubsection`. Was passiert?
- Füge am Anfang des Dokuments die Anweisung `\tableofcontents` ein. Was passiert beim Übersetzen? Welche Dateien entstehen?
- Übersetze die  $\text{\LaTeX}$ -Datei ein zweites Mal. Was hat sich verändert? Erkläre Deine Beobachtungen.

# Aufgaben

- Füge vor dem Inhaltsverzeichnis folgende Zeilen ein:

```
\title{Hello World}  
\author{Joe User}  
\maketitle
```

Welche Informationen erscheinen jetzt im gesetzten Dokument?

- Lösche die aux- und die toc-Datei, und kommentiere die Zeile mit dem Babel-Paket aus:

```
%\usepackage[german]{babel}
```

Was ändert sich?

# Dokumentklassen

- original  $\text{\LaTeX}$ : `article`, `report`, `book`
- KOMA-Skript: `scrartcl`, `scrreprt`, `scrbook`
- Gliederungsstufen:

|                      |                       |                       |                          |                  |
|----------------------|-----------------------|-----------------------|--------------------------|------------------|
| <code>article</code> |                       | <code>\section</code> | <code>\subsection</code> | <code>...</code> |
| <code>report</code>  | <code>\chapter</code> | <code>\section</code> | <code>\subsection</code> | <code>...</code> |
| <code>book</code>    | <code>\chapter</code> | <code>\section</code> | <code>\subsection</code> | <code>...</code> |

- `...`, `\subsubsection`, `\paragraph`, `\subparagraph`
- zusätzlich `\part`
- mehrere `article` können ein `book` bilden

# Aufgaben

- Ersetze die Dokumentklasse `article` durch `scrartcl`. Wie ändert sich die Formatierung?
- Definiere mit der Anweisung `\chapter` ein oder mehrere Kapitel und probiere die Dokumentklassen `report`, `scrreprt`, `book` und `scrbook` aus.

# Aufgaben

- Erweitere die  $\text{\LaTeX}$ -Datei folgendermaßen:

```
\maketitle
\frontmatter
\tableofcontents
\mainmatter
  \chapter{...}
  \section{...}
  \chapter{...}
  \section{...}
\appendix
  \chapter{...}
  \chapter{...}
```

Welchen Effekt haben die zusätzlichen Anweisungen?