

# $\text{\LaTeX}$ für Einsteiger

Erste Sitzung

Valentin Heinz

05.10.2015

# Vorstellung

- ▶ herzlich willkommen! :)
- ▶ Valentin Heinz ([Valentin.Heinz@hhu.de](mailto:Valentin.Heinz@hhu.de))
- ▶ Computerlinguistik im Master (Abschlussarbeit fehlt noch)
- ▶ ich mag L<sup>A</sup>T<sub>E</sub>X und Gestaltung
- ▶ ich unterrichte und programmiere gerne

# Praktisches, Fragen

- ▶ Sie oder Du?
- ▶ haben Sie einen Laptop dabei? Haben Sie Strom?
- ▶ wann machen wir Mittagspause?

# Überblick

- ▶ der heutige Inhalt besteht aus folgenden Teilen:
  - ▶  $\text{\LaTeX}$ -Geschichte
  - ▶  $\text{\LaTeX}$ -Funktionsweise
  - ▶  $\text{\LaTeX}$ -Dokumentstruktur
  - ▶ mehr Befehle
  - ▶ Schriftformatierung
  - ▶ mehr Struktur
  - ▶ Übungen
- ▶ wie sieht die Struktur eines Teils aus?
  - ▶ Einstieg: Wiederholung, Fragen
  - ▶ Inhalte werden idealer Weise durch den Theorieteil, Praxisteil und Übungen vermittelt

# Motivation

- ▶ oder: warum gebe ich diesen Kurs?
- ▶ es gibt viele Übersichten, Bücher, Webseiten usw. zu  $\text{\LaTeX}$ , aber kaum praktische Kurse
- ▶ in Seminaren sind die Lernerfolge gut überprüfbar
- ▶ die Hilfestellung ist direkt und passend
- ▶ Erfahrungsaustausch, Tipps & Tricks bündeln

# Vorstellung 2

- ▶ Fachsemester
- ▶ Vorstellungsrunde:
  - ▶ wie heißen Sie?
  - ▶ ihre nächste große Arbeit ist ... ?
  - ▶ welche fachliche Ausrichtung haben Sie?
  - ▶ welche Schwerpunkte haben Sie?

# Dokumenttypen

- ▶ Bedarfsevaluation
- ▶ welche Dokumenttypen sind für Sie relevant?
  - ▶ Buch
  - ▶ Artikel/Hausarbeit
  - ▶ Präsentation
  - ▶ Magazin
  - ▶ Konferenzposter
  - ▶ Bewerbungsunterlagen
  - ▶ Lebenslauf
  - ▶ Umfragebogen
  - ▶ Brief
  - ▶ Gedicht, Notensatz, Theaterstück, ...

# Ziele

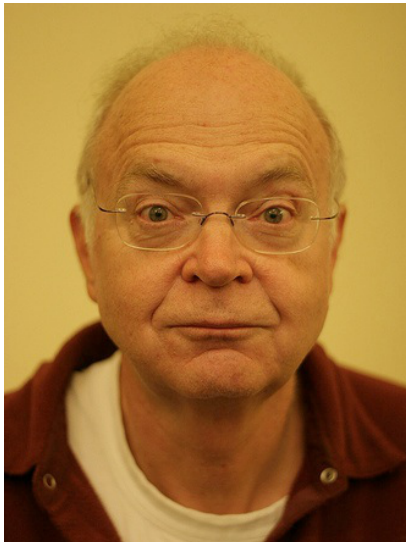
- ▶  $\text{\LaTeX}$  lernen und dann **üben**
- ▶ einfacheres, schöneres und effizienteres wissenschaftliches Arbeiten mit  $\text{\LaTeX}$
- ▶ Probleme mit  $\text{\LaTeX}$  selber lösen
- ▶ einen Überblick über die Möglichkeiten und das Ökosystem bieten
- ▶ Fragen richtig stellen, Hilfe zur Selbsthilfe



# Eigenleistung

- ▶ selbsterstelltes Cheatsheet
- ▶ eigenständig erstellte Arbeit, diese beinhaltet:
  - ▶ eine aussagenlogische Formel
  - ▶ eine Tabelle mit einer Kopfzeile und mindestens drei normalen Zeilen
  - ▶ etwas Fließtext
  - ▶ ein Zitat mit Quellenangabe
  - ▶ eine Besonderheit deines Spezialgebiets (z.B. phonetische Transkription)
  - ▶ das Literaturverzeichnis
  - ▶ wichtig: Kommentare, welcher Teil des Quelltextes dem jeweiligen obigen Teil entspricht
- ▶ keine Panik, folgendes sieht nach mehr aus, als es ist! Dennoch stelle ich damit sicher, dass Sie mit  $\text{\LaTeX}$  arbeiten können.

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



# Donald E. Knuth

- ▶ emeritierter Professor (Stanford)
- ▶ fast jährliche Preise und Auszeichnungen seit 1970 bis heute
- ▶ 25 Ehrendokortitel
- ▶ Knuth-Preis: außergewöhnliche Leistungen in den Grundlagen der Informatik
- ▶ Asteroid: (21656) Knuth

# Trivia?

- ▶ ab 1. Januar 1990: mehr Konzentration auf Arbeit, nutzt keine E-Mails mehr
- ▶ Korrektheit ist sehr wichtig: verschickt Schecks über 2.56\$ für gefundene Fehler
- ▶ (wg. der Angst vor Scheckbetrug benutzt er nun die digitale Währung einer fiktiven Bank, zahlt das Guthaben aber auch gerne aus)

verstehen Sie Spaß?



# Ein Buch über Comiler

- ▶ Knuth (noch Student im Hauptstudium) schreibt ein Buch über Compiler
- ▶ *Do you mind if I make this book a little bit longer, because I think there's a need for explaining these things in somewhat more detail.*
- ▶ *They said, ,Oh no, go right ahead. Make it as long as you feel necessary.'*
- ▶ Ergebnis: 3900 Seiten (1968)

# The Art of Computer Programming

- ▶ Band 1: 1968, Band 5 (von 7): 2020
- ▶ legendäres Grundlagenwerk
- ▶ eigener theoretischer, idealer Computer MIX, eigene Assemblersprache MIXAL
- ▶ Themen: Algorithmen und Datenstrukturen (und Zahlentheorie, Suchen, Sortieren, uvm.)

# T<sub>E</sub>X entsteht

- ▶ Überarbeitung von Band 2: Knuth erfindet T<sub>E</sub>X
- ▶ Schriftartendesign, Satztechnik, ...
- ▶ Bücher über T<sub>E</sub>X in T<sub>E</sub>X: Spezifikation und Benutzung
- ▶ Erklärung von Verfahren, z.B.: Worttrennungsalgorithmus



# $\text{\LaTeX}$ und $\text{\LaTeX} 2_{\epsilon}$

- ▶ 1980: Leslie Lamport entwickelt Makros für  $\text{\TeX}$ : *LaTeX*
- ▶ 1990: Lamport beendet die Entwicklung.  $\text{\LaTeX}$  wird fortgeführt.
- ▶ 1994:  $\text{\LaTeX} 2_{\epsilon}$  ist die Fortführung von  $\text{\LaTeX}$
- ▶ bedeutet:  $\text{\LaTeX}$  ist älter als Unicode

# Exkurs: Unicode

- ▶ internationaler Standard zur Zeichenkodierung (Version 1.0.0: Oktober 1991)
- ▶ soll *jedes* sinntragende Zeichen umfassen
- ▶ alle bekannten Schriftkulturen und Zeichensysteme
- ▶ ein Beispiel für einen Codepunkt: U+1F600 (welcher ist das?)
- ▶ UTF-8: 8-bit Kodierung für Unicode-Zeichen, sehr verbreitet
- ▶ Internet Engineering Task Force: neue Protokolle müssen UTF-8 können

# Aufgaben

- ▶ mit dem Hintergrund von  $\text{\LaTeX}$  und Unicode-Wissen, versuchen Sie folgendes herauszufinden:
- ▶ was ist pdfTeX?
- ▶ was ist XeTeX? Inwiefern ist XeTeX besser als  $\text{\LaTeX}$ ?
- ▶ was ist LuaTeX? Was macht LuaTeX anders als pdfTeX?

Pause

Pause?

# WYSIWYG

- ▶ What You See Is What You Get
  - ▶ Beispiele: Word/OpenOffice/Libreoffice
  - ▶ Benutzer: Inhalt eingeben → Formatierung
  - ▶ Programm: nach Umgebung → Layouting der Eingabe
  - ▶ Prozess: nach und nach
- ▶ man legt fest, wie eine Überschrift *aussieht*
- ▶ man legt *nicht* fest, was (strukturell!) eine Überschrift *ist*
- ▶ (aber: Formatvorlagen)

# WYSIWYM

- ▶ What You Get Is What You Mean
- ▶ Auszeichnungssprachen: *logische* Auszeichnung von *Strukturen*
- ▶ Beispiel: HTML
  - ▶ Titel: `<title>Ich bin der Titel</title>`
  - ▶ kursiv: `<i>ich bin kursiv</i>`
- ▶ Beispiel:  $\text{\LaTeX}$ 
  - ▶ Titel: `\title{Ich bin auch ein Titel}`
  - ▶ kursiv: `\textit{und ich bin auch kursiv}`

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

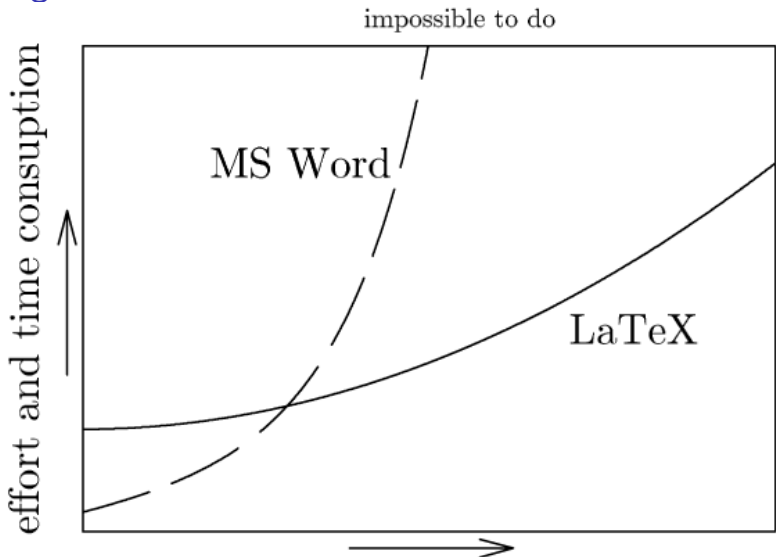
- ▶ man legt fest, was vom Text z.B. eine Überschrift ist
- ▶ man legt aber *nicht* fest, wie eine Überschrift aussieht
- ▶ Berufe: Typograph, Schriftsetzer, Designer
- ▶ Vorgang: erst schreiben (Quelltext) und auszeichnen, dann layouts *lassen*
- ▶ kompilieren: Übersetzung vom Quelltext zum fertigen Dokument

# gefühlte Vorteile

- ▶ hochwertige Typographie (was bedeutet das??)
- ▶ Dokumente sind leicht unterteilbar in einzelne Dateien
- ▶ einfache Arbeit mit Referenzen und Zitierstilen
- ▶ Sachen die in Word nicht funktioniert haben
  - ▶ Bsp.: Pragmatikhandout: mehrfache Potenz.  $\text{\LaTeX: } \backslash(a^{b^{c^d}})$
- ▶ einfaches Aufschreiben von beliebig komplexen mathematischen Formeln
- ▶ Text (besonders UTF-8) ist toll, weil universell und immer zugänglich



# Einarbeitungszeit



# Vorteile, die man später bemerkt

- ▶ Skalierbarkeit: „Bücher schreibt man in  $\text{\LaTeX}$ “
- ▶ kostenlos und bleibt es
- ▶ Freiheit: keine Hersteller- bzw. Lizenzbindung (MS-Office-Preise, SPSS-Preise)
- ▶ Akzeptanz & Verbreitung
  - ▶ kollaboratives Arbeiten ist leichter (jeder kann seinen Teil unabhängig einbinden)
  - ▶ (Abschluss)arbeiten in der Mathematik, Naturwissenschaft und Technik
  - ▶ Paper:  $\text{\LaTeX}$ -Templates von Journals

# Nachteile

- ▶ schwieriger Einstieg
- ▶ steilere Lernkurve
- ▶ ... dafür gibt es dieses Seminar! Schildert mir Nachteile, die mir nicht einfallen)
- ▶ Schriftarten (dafür gibt es aber  $X_{\exists}T_{\mathbb{E}}X$  und Lua $T_{\mathbb{E}}X$  usw.)

Pause

Pause?



# Editor

Um erste  $\text{\LaTeX}$ -Erfahrungen zu machen

- ▶ rufen Sie <https://www.overleaf.com/> auf
- ▶ dort schalten Sie links oben auf Source
- ▶ dann schalten Sie rechts oben den Preview-Schalter auf Manual
- ▶ dann löschen Sie das Dokument

# erste (!) Dokumenttypen

Sie müssen bei jedem  $\text{\LaTeX}$ -Dokument den Dokumenttypen angeben:

- ▶ `article` Journals, usw.
- ▶ `report` längere Berichte, kurze Bücher, Diplomarbeiten
- ▶ `book` richtige Bücher
- ▶ `letter` Briefe
- ▶ `beamer` Präsentationen (siehe  $\text{\LaTeX}$ /Präsentationen)

Weiterhin müssen Sie definieren, wo unser Dokument anfängt.

# Minimalbeispiel (Struktur)

```
\documentclass{article}
```

```
\begin{document}
```

```
    Hello World!
```

```
\end{document}
```

- ▶ Dokumentaufbau (Nicht-Dokument, Dokument, Text)
- ▶ Frage: was ist was?



# Syntax

- ▶ Ab und an werde ich die Befehlssyntax näher besprechen, wie in diesen Beispielen

```
\befehl{PARAMETER}
```

```
\begin{PARAMETER}
```

```
\end{PARAMETER}
```

# Keine Probleme?

- Finden Sie Probleme in diesem Beispiel?

```
\documentclass{article}
```

```
\begin{document}
```

Außerdem und überhaupt: „wir“ wollen mehr!

```
\end{document}
```

# Probleme: Umlaute und Sonderzeichen

- ▶ Anführungszeichen
- ▶ für deutschsprachige Arbeiten unübliche Ränder und Abstände
- ▶ keine Kommentare

# Dokumenttypen des Koma-Script-Pakets (Markus Kohm)

- ▶ `scrartcl` Journals, usw.
- ▶ `scrreprt` längere Berichte, kurze Bücher, Diplomarbeiten
- ▶ `scrbook` richtige Bücher
- ▶ `scrlettr2` Briefe
- ▶ `beamer` wer weiß das noch?

# Ein besseres Beispiel

- ▶ besuchen Sie die Seminarseite:
- ▶ <https://github.com/inktrap/LaTeXKurs>
- ▶ laden Sie: 01-hello-world.tex herunter.
- ▶ kopieren Sie den Inhalt in overleaf

# Bestandteile

- ▶ Kommentare
- ▶ Encoding definieren: UTF-8 (ISO 8859-1)
- ▶ Spracheinstellung: babel
- ▶ Pakete einbinden. Syntax:  
`\usepackage[OPTIONEN]{PAKETNAME}`
- ▶ Paketinformationen: <https://www.ctan.org/>

# Änderungen

- ▶ was passiert, wenn Sie einen anderen Dokumenttypen wählen?
- ▶ ihre Erwartung: wie würden Sie einen Absatz einfügen? Was passiert?
- ▶ binden Sie das Packet `ellipsis` ein und finden Sie heraus, was es macht ...
- ▶ wie kann man die neue Rechtschreibung und Din-A4 verwenden? Hinweis: Dokumenttyp

# Übungen

- ▶ Methode: Pairprogramming
- ▶ eine Person programmiert (Driver), die andere schaut zu und denkt mit (Navigator)
- ▶ der Navigator denkt auch eigenständig über das Problem nach
- ▶ versteht der Navigator einen Schritt des Drivers nicht, kann er direkt nachfragen
- ▶ steht der Driver vor einem Problem, kann er dieses dem Navigator beschreiben



# Paarprogrammierung 1: Durcheinander

- ▶ laden Sie `01-hello-world_wrong-order.tex`
- ▶ erstens: Durcheinander sortieren, dann
- ▶ zweitens: damit das nicht wieder passiert, erweitern Sie das Template um Kommentare

## Paarprogrammierung 2: Fehler

- ▶ tauschen Sie die Rollen
- ▶ laden Sie `01-hello-world_wrong-cmd.tex` herunter
- ▶ erklären Sie im Gespräch/zusammen mit Ihrem Nachbarn die Fehlermeldung(en)
- ▶ korrigieren Sie alle syntaktischen Fehler

Pause

Pause?

# Zusammenfassung der Syntax

# Syntax

- ▶ PARAMETER sind Argumente
- ▶ PARAMETER sind nicht immer OPTIONEN
- ▶ OPTIONEN sind optionale PARAMETER
- ▶ unser erster Befehl: `\documentclass{}`. Syntax:  
`\documentclass[OPTIONEN]{PARAMETER}`
- ▶ unsere erste Umgebung: durch `\begin{}` und `\end{}`. Syntax:

# Befehlsarten

% nicht optional  
\befehl{PARAMETER}

% optional  
\befehl[OPTIONEN]

% beides  
\befehl[OPTIONEN]{PARAMETER}

# Textformatierung

- ▶ Die Textformatierung findet mit einem Schalter statt.
- ▶ Dieser nimmt keine Parameter und bleibt bei Größenangaben aktiv, bis ein anderer Schalter aktiv wird.
- ▶ Hinweis: `\LaTeX`, etc. ist kein Schalter, sondern nur ein Textbefehl

Syntax:

```
% schalter ist nicht aktiv  
\schaltername  
% schalter ist aktiv
```

# Größe

- ▶ `\tiny`
- ▶ `\scriptsize`
- ▶ `\footnotesize`
- ▶ `\small`
- ▶ `\normalsize`
- ▶ `\large`
- ▶ `\Large`
- ▶ `\LARGE`
- ▶ `\huge`
- ▶ `\Huge`



# Übung

- ▶ betrachten Sie folgende Geschichte:
- ▶ tiny and huge are walking to the small house of footnotesize. tiny says: "this is too Huge for me" and huge thinks thats not normalsize either
- ▶ formatieren Sie das Wort der ersten Größe mit der letzten vorkommenden Größe.
- ▶ d.h.: tiny mit normalsize, huge mit huge, usw.

# Größe als Umgebung

- ▶ die Größe kann auch per Umgebung definiert werden:  
`\begin{scriptsize}` Ein kleiner Satz.  
`\end{scriptsize}` Ich bin wieder normal.
- ▶ dies ist häufig viel leichter zu überschauen, da man die Umgebung bewusst beendet.

# Schriftformatierung und Übung

Übernehmen Sie folgende Formatierungsangaben in Ihr Cheatsheet:

Laut

```
\textsc{Gaius Iulius Caesar} sollte man neue Begriffe  
\textit{kursiv} schreiben und nicht  
\textbf{fett}, besonders  
\texttt{Schreibmaschinen} sind eher  
\texttt{\textbf{\textit{total}}} unnötig.
```

# Hausaufgabe: Farben

- ▶ finden Sie heraus, wie sie Text farbig gestalten können
- ▶ zeigen Sie dies morgen an einem Beispiel

Pause

Pause?

# Kapitel

```
% 1  
\section[ALTERNATIV]{ABSCHNITT}
```

```
% 2  
\section*{ABSCHNITT}
```

- ▶ tragen Sie etwas für ABSCHNITT ein.
- ▶ ALTERNATIV ist die Kurzform, die auch z.B. im Inhaltsverzeichnis auftaucht
- ▶ wir haben bereits gesehen, dass eine [OPTION] auch weggelassen werden kann.
- ▶ was bewirkt das \* Sternchen in 2?

# Unterkapitel

- wir wollen nun Abschnitte und Unterabschnitte

```
% 3  
\section[ALTERNATIV]{ABSCHNITT}
```

```
% 4  
\subsection[ALTERNATIV]{UNTERABSCHNITT}
```

# Übungen

- ▶ wie würden sie den Unterabschnitt von einem Unterabschnitt bilden?
- ▶ wie lange funktioniert das? Was passiert dann?



# Nummerierte Listen

```
% 5
\begin{enumerate}
  \item ITEM 1
  \item ITEM 2
  \item ITEM 3
\end{enumerate}
```

# Nicht nummerierte Listen

```
% 6
\begin{itemize}
  \item ITEM 1
  \item ITEM 2
  \item ITEM 3
\end{itemize}
```

# Übungen

- ▶ was passiert, wenn Sie ein `\end{enumerate}` oder `\end{itemize}` vergessen?
- ▶ was passiert, wenn Sie ein `\begin{enumerate}` oder `\begin{itemize}` vergessen?
- ▶ kann ein `\item` auch alleine stehen? Also: nicht in einer Liste?
- ▶ was ist, wenn eine Liste kein `\item` hat?
- ▶ was passiert, wenn Sie nummerierte Listen schachteln? Wenn das geht, wie weit geht das?
- ▶ was passiert, wenn Sie nicht-nummerierte Listen schachteln? Wenn das geht, wie weit geht das?
- ▶ erinnern Sie sich an `\section*{Nicht-nummerierte Überschrift}`? Geht dies auch bei nummerierten Listen?

# Blockzitate

- ▶ Blockzitate sind auch nur Umgebungen, genau wie `document`, `enumerate`, `itemize`, ...

```
\begin{quote}  
Ich bin ein Blockzitat  
\end{quote}
```

# Ende des ersten Tages

- ▶ vielen Dank für Ihre Aufmerksamkeit!
- ▶ haben Sie Fragen?
- ▶ wir üben nun noch!