

L^AT_EXKurs

Eine Zusammenfassung

Marvin Raiser

12. Juni 2021

Inhaltsverzeichnis

1	Einleitung	4
1.1	Geschichte	4
1.2	Wofür ist \LaTeX geeignet?	4
1.3	Wie funktioniert \TeX ?	4
1.3.1	What You See Is What You Get	4
1.3.2	What You See Is What You Mean	4
1.3.3	Kompilieren einer \LaTeX Datei	4
1.4	\TeX -Struktur	5
1.5	Das erste Dokument	5
2	Dokumentheader	7
2.1	Dokumentklassen	7
2.1.1	Typische Klassen	7
2.1.2	KOMA-Klassen	7
2.1.3	Wichtige <code>documentclass</code> Optionen	7
2.2	Titel	7
2.3	Wichtige Pakete	7
3	Dokumentstruktur	8
3.1	Abschnitte und Überschriften	8
3.2	Umgebungen	8
3.3	Verzeichnisse	8
3.4	Ausrichtung	8
3.5	Abstände	8
3.6	Zeilen- und Seitenumbrüche	8
4	Text	9
4.1	Schriftart	9
4.2	Schriftgröße	9
4.3	Symbole	9
4.4	Trennzeichen und Striche	9
4.5	Datum	9
5	Listen	10
5.1	Auflistungen	10
5.2	Auflistungen verschachtelt	10
5.3	Auflistungen mit andere Zeichen	10
5.4	Auflistungen immer mit anderen Zeichen	10
5.5	Aufzählungen	10
5.6	Aufzählungen verschachtelt	10
5.7	Aufzählungen mit <code>enumitem</code> packet	11
6	Abbildungen	12

7 Tabellen	13
8 Mathe	14
8.1 Maxwellgleichung	14
9 Fließumgebungen und Verweise	15
9.1 Fließumgebungen	15
9.2 Referenzieren	15
9.3 Abbildungen	15
9.4 Tabellen	15
9.5 Mathematik	15
9.6 Fußnoten	15
10 Große Projekt Strukturen	16
10.1 Auslagern durch <i>input</i> und <i>include</i>	16
10.2 Dateistruktur	16
10.3 Versionsverwaltung	16
11 Weiterführende Literatur	16
12 Danksagung	17

1 Einleitung

1.1 Geschichte

1977 entwickelte *Donald E. Knuth* das Textverarbeitungssystem TeX (griechisch texum). *Leslie Lamport* erweiterte TeX mit mehr Abstraktion, starker Vereinfachung und einem spezifischeren Befehlssatz zu LaTeX. Ursprünglich als Darstellungswerkzeug mathematischer Formeln gedacht wurde es schnell auch von zahlreichen Autoren zum einheitlichen Layouten ihrer Werke genutzt. Darauf aufbauend werden bis heute zahlreiche LaTeX Erweiterungen entwickelt und der Einsatz in Wissenschaft und des Schreibens ist unabdingbar.¹

1.2 Wofür ist L^AT_EX geeignet?

Gut geeignet ist L^AT_EX aufgrund des einheitlichen hochkonfigurierbaren Layouting, mathematischen Werkzeuge, Mehrsprachigkeit und Bibliographieerstellung für Schriftstücke mit logischem Aufbau wie z.B. Naturwissenschaftliche Arbeiten, Geisteswissenschaftliche Arbeiten, Artikel, Abschlussarbeiten, sowie Bücher und simple Präsentationen.

Nicht geeignet ist L^AT_EX für Dokumente mit wenig Struktur, Präsentationen (bunt, drehend, blinkend, Animationen), Plakate, Dokumente mit vielen uneinheitlichen Bildern.

1.3 Wie funktioniert T_EX?

1.3.1 What You See Is What You Get

Bei klassischen Texteditoren wie Word schreibt und sieht man genau das was man will und bekommt. Formatierung und Positionierung passiert dabei im Hintergrund automatisch und sieht der User bereits verarbeitet. Das erleichtert das Schreiben, erschwert jedoch auch das Layouting von großen Dokumenten, da die Eigenschaften intransparent sind.

1.3.2 What You See Is What You Mean

Dem *WYSIWYG* steht das *What You See Is What You Mean (WYSIWYM)* entgegen. Dabei wird grundsätzlich zwischen den Textdateien und dem verarbeiteten Ergebnis unterschieden. Die Textdateien sind jedoch mit der zugrunde liegenden Sprache L^AT_EX geschrieben (vergleichbar mit einer Programmiersprache). Diese Sprache abstrahiert Layout, Text und sonstige Medien und bringt diese in eine logische Struktur.

1.3.3 Kompilieren einer L^AT_EX Datei

Diese Sprache und Struktur muss zur gegebenen Zeit verarbeitet und zusammengelegt bzw. *kompiliert* werden, um daraus das gewünschte Resultat, zu generieren. Wie in illustriert sind mehrere Programme zum Kompilieren eines Tex Dokuments notwendig. Diese

¹Aus https://www.selflinux.org/selflinux/html/latex_geschichte01.html

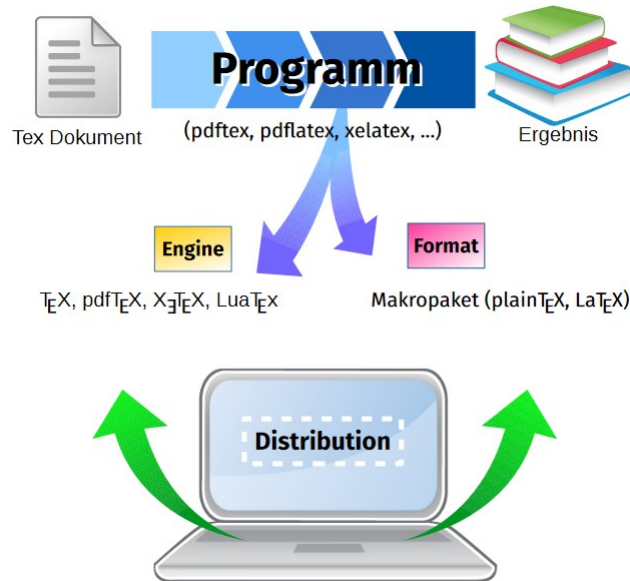


Abbildung 1: Wie Tex Dokumente kompiliert werden

Programme setzen sich aus der generellen \TeX Engine und weiteren Makropaketen zusammen die wiederum unterschiedliche Formatierungen reinbringen. Distributionen sind dabei unterschiedliche Weiterentwicklungen in unterschiedliche Richtungen und Systeme, die aber alle auf der selben Basis aufbauen. In der Regel bringt eine Distribution alle typischen Pakete mit die man brauchen wird. ²

1.4 TeX-Struktur

Ein Tex Projekt ist stets zwei geteilt. Ein *Header* definiert den Dokumenttypen, setzt Einstellungen zum Layout und konfiguriert die Pakete die man braucht. Im *Body* oder auch *document* befindet sich dann der gesamte Inhalt des Dokumentes. Der *Header* wird einmalig am Anfang des Projekts gesetzt, damit das Layout und die wichtigen Einstellungen getroffen sind, bevor man sich im Anschluss auf den Inhalt fokussiert. Die Arbeit konzentriert sich daher stark auf den eigentlichen Inhalt im *document*.

1.5 Das erste Dokument

Im Folgenden verwenden wir *texlive* als Distribution und *xelatex* als Compiler. Als Editor ist TextStudiu zu empfehlen. Alternativ kann man auch VS Code mit der Extension *LaTeX Workshop* verwenden.

²Weitere Distributionen und Programme: <http://www.tug.org/interest.html>

Beim Erstellen der ersten Datei darauf achten, dass alle Tex Dateien auf *.tex* enden.

```
\documentclass{minimal}  
\begin{document}  
Hallo Welt!  
\end{document}
```

Der *Header* definiert hier eine *documentclass* und setzt diese auf *minimal*, dazu später mehr.

Leicht zu erkennen leitet *begin* die *document* Umgebung, unseren *Body* ein. Nachdem dort der Text *Hallo Welt!* gedruckt wird, schließt sich das *document* auch wieder.

Super! Das erste minimale Dokument steht und ist einsatzbereit!

2 Dokumentheader

Im *Header* werden einmalig globale Einstellungen getroffen, um unter anderem einheitliches Design zu garantieren oder Pakete zu konfigurieren. Eingeleitet wird der *Header* durch *documentclass* und geschlossen durch *begin(document)*

2.1 Dokumentklassen

Dokumentklassen legen erste allgemeine Layoutregeln fest wie beispielsweise Ränder, Abstände, Schriftgrößen etc.

2.1.1 Typische Klassen

Die typischen Klassen nach Relevanz und Häufigkeit sortiert. Meist wird *article* verwendet.

article	(Kurze) Artikel
report	Reporte, Tagungsberichte
book	Bücher
letter	Briefe
minimal	Minimalbeispiele
beamer	Präsentationen

Tabelle 1: Die typischen Dokumentklassen

2.1.2 KOMA-Klassen

KOMA-Klassen sind moderne Erweiterungen der typischen Klassen. Sie modernisieren nicht nur das Layout und Design, sondern erlauben es einfach KOMA-Skripte zu nutzen, wodurch mehr Funktionalität und Pakete unterstützt werden.

scrartcl	Erweiterung von article
scrreprt	Erweiterung von report
scrbook	Erweiterung von book
scrlettr2	Sehr mächtige Briefklasse

Tabelle 2: Zusätzliche KOMA-Klassen

2.1.3 Wichtige *documentclass* Optionen

2.2 Titel

2.3 Wichtige Pakete

3 Dokumentstruktur

3.1 Abschnitte und Überschriften

3.2 Umgebungen

3.3 Verzeichnisse

3.4 Ausrichtung

3.5 Abstände

3.6 Zeilen- und Seitenumbrüche

4 Text

4.1 Schriftart

4.2 Schriftgröße

4.3 Symbole

4.4 Trennzeichen und Striche

4.5 Datum

5 Listen

5.1 Auflistungen

- Apfel
- Birne

5.2 Auflistungen verschachtelt

- Obst
 - Apfel
 - Birne
- Gemüse

5.3 Auflistungen mit andere Zeichen

- Apfel
- Birne
- Erdbeere

5.4 Auflistungen immer mit anderen Zeichen

- Apfel
- Birne

5.5 Aufzählungen

1. Apfel
2. Birne

5.6 Aufzählungen verschachtelt

1. Obst
 - a) Apfel
 - b) Birne
2. Gemüse

5.7 Aufzählungen mit enumitem packet

- (a) Apfel
- (b) Birne
- (c) Banane



Abbildung 2: Ein süßer Pinguin in Fließumgebung

6 Abbildungen

In \LaTeX können auch Bilder eingefügt werden. Hierfür kann *includegraphics* genutzt werden.



Doch oftmals sollen Bilder flüssig und dynamisch dem Text angepasst werden. Mit der Fließumgebungen *figure* kann hierbei einem Bild eine *caption* und ein *label* gegeben werden. Das Label kann im Text referenziert werden.

Schaut euch den süßen Pinguin in Abbildung 2 an! Doch warum befindet sich dieser so komisch oben? \LaTeX versucht hier die Abbildung an den restlichen Inhalt der Seite anzupassen und platziert daher das Bild so komisch an den Anfang. Mit der *[h]* Option der *figure* Fließumgebung kann erzwungen werden, dass die Abbildung an der jetzigen Position eingefügt werden soll. Mit *centering* kann der nachfolgende Inhalt, die Abbildung, zentriert werden.



Abbildung 3: Pinguin Familie, awwww

7 Tabellen

a	b	c
A	B	C

Tabelle 3: Das einfache abc

Im Text kann man auf Tabelle 3 verweisen

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

8 Mathe

Im *inline* Mathematik-Modus definieren wir die Masse m , Energie E sowie die Lichtgeschwindigkeit c , die wir anschließend im *display* Mathematik-Modus einer Formel verwenden

$$E = mc^2$$

Auch Variablen und griechische Symbole sind möglich

$$\alpha = \theta \cdot \gamma$$

Brüche können mit *frac* beschrieben werden

$$\alpha = \frac{\beta \cdot \alpha}{\Gamma}$$

Im Gegensatz zum einfachen *display* Mathematik-Modus sind Gleichungen in der *equation* Umgebung nummeriert.

$$\int_0^\infty f(x) \, dx \tag{1}$$

Mit *align* kann man Formeln schön untereinander alignen.

$$a = b + c \tag{2}$$

$$c + d = e + f \tag{3}$$

8.1 Maxwellgleichung

In cgs-Einheiten und differentieller Form lauten die vier Maxwellgleichungen:

$$\nabla \cdot \vec{E} = 4\pi\rho \qquad \text{Gaußsches Gesetz} \tag{4}$$

$$\nabla \cdot \vec{B} = 0 \tag{5}$$

$$\nabla \times \vec{E} = -\partial_{ct}\vec{B} \qquad \text{Faradaysches Induktionsgesetz} \tag{6}$$

$$\nabla \times \vec{B} = \frac{4\pi}{c}\vec{j} + \partial_{ct}\vec{E} \qquad \text{Ampère-Maxwellsches Durchflutungsgesetz} \tag{7}$$

9 Fließumgebungen und Verweise

9.1 Fließumgebungen

Layout, Positionierung, label, caption

9.2 Referenzieren

ref, eqref, pageref

9.3 Abbildungen

Am Beispiel für eine Abbildung

9.4 Tabellen

Am Beispiel für eine Tabelle

9.5 Mathematik

Am Beispiel für eine Formel

9.6 Fußnoten

10 Große Projekt Strukturen

10.1 Auslagern durch *input* und *include*

10.2 Dateistruktur

10.3 Versionsverwaltung

11 Weiterführende Literatur

12 Danksagung

Diese L^AT_EX Zusammenfassung ist während dem L^AT_EX Kurs des Heidelberger Life-Science Labs am 12. Juni 2021 entstanden.

Großes Dankeschön an das Kursleitungsduo Hannes Keppler und Jakob Kreft!